OPTICON

Electronic Shelf Label
ESL Server manual

The ESL server manual describes how to demonstrate and develop systems using the Opticon ESL solution.

User Manual
v1.49.0

All information in this manual is subject to change without notice.

Document History

Model Number:	ESL Server Manual	**Specification Number:**	
Edition:	1.49.1	**Original Spec Number:**	
Date:	2015		

Copyright 2015 Opticon. All rights reserved.

Limited Warranty and Disclaimers

PLEASE READ THIS MANUAL CAREFULLY BEFORE INSTALLING OR USING THE PRODUCT.

Serial Number

A serial number appears on all Opticon products. This official registration number is directly related to the device that is purchased. Do not remove the serial number from your Opticon device. Removing the serial number voids the warranty.

Warranty

Unless otherwise agreed in a written contract, all Opticon products are warranted against defects in materials and workmanship for two years after purchase. Opticon will repair or, at its option, replace products that are defective in materials or workmanship with proper use during the warranty period. Opticon is not liable for damages caused by modifications made by a customer. In such cases, standard repair charges will apply. If a product is returned under warranty and no defect is found, standard repair charges will apply. Opticon assumes no liability for any direct, indirect, consequential or incidental damages arising out of use or inability to use both the hardware and software, even if Opticon has been informed about the possibility of such damages.

Packaging

The packing materials are recyclable. We recommend that you save all packing material to use should you need to transport your scanner or send it for service. Damage caused by improper packaging during shipment is not covered by the warranty.

Trademarks

Trademarks used are the property of their respective owners.

Opticon Inc. and Opticon Sensors Europe B.V. are wholly owned subsidiaries of OPTOELECTRONICS Co., Ltd., 12-17, Tsukagoshi 4-chome, Warabi-shi, Saitama, Japan 335-0002. TEL +81-(0) 48-446-1183; FAX +81-(0) 48-446-1184

SUPPORT

USA	Europe
Phone: 800-636-0090	Phone: +31 235692728
Email: support@opticonusa.com	Email: support@opticon.com
Web: www.opticonusa.com	Web: www.opticon.com

Contents

1. Features and Specifications

1.1. Abstract

This manual provides a preliminary description on how to set-up a customized Electronic Shelf Labels (ESL) project with e-Paper ESLs from Opticon Sensors Europe BV.

1.2. Contents of an ESL demo kit

The ESL demo kit consists of the following components:

- Ethernet ESL base station (EBS-30)
- 6 2" e-paper ESLs (EE200 or EE201) including 6x2 CR2450 batteries
- UTP cable
- 6.0V 2A Power supply (European)
- PC Installation Software + examples
- Opticon ESL Server Kit User Manual

The demo kit does NOT include a barcode terminal, which can be used to link ESLs to products using barcodes. Any of the following Opticon Barcode Terminals can be used for this purpose:

- OPH100x (+ CRD100x cradle)
- H13 (+ CRD13 cradle)
- OPL9728 (+ CRD9723RU cradle)
- OPN2002/3/4/5/6 (+ mini USB cable)
- CLK3000 (+ mini USB cable)
- H21 / H22 (using Wifi)

1.3. Physical Features

1.3.1. Dimensions

2" e-paper ESL (EE200):   W 70.0 x D 32.0 x H 15.0 mm

2" e-paper ESL (EE201):   W 65.0 x D 34.0 x H 15.0 mm

Base station (EBS-30):   W 145.0 x D 110.0 x H 30.0 mm (excl. antenna)

1.3.2. Weight

2" e-paper ESL (EE200/1):   25 g (incl. 2 batteries)

Base station (EBS-30):   125 g (excl. power supply)

1.4. Environmental Specifications

1.4.1. Operating Temperature and Humidity (EE200/1)

Temperature: 5 to 50° C

Humidity: 20-85%

1.4.2. Storage Temperature and Humidity (EE200/1)

Temperature: -20 to 60° C

Humidity: 20-85%

1.5. Interface Specifications

The radio interface of both the base stations and ESLs have the following specifications:

- 2.4gHz IEEE 802.15.4 compatible
- Transmission speeds: up to 250 kbps

1.6. Hardware Specifications

Items	Specifications		Remarks
CPU	Type	Stellaris LM3S9B96 (EBS-30) CC2530-F128 (EE200) CC2533-F96 (EE201)	
	Internal ROM	256 KB (EBS-30) 128 KB (EE200) 96 KB (EE201)	
	Internal RAM	96KB (EBS-30) 8 KB (EE200) 6 KB (EE201)	
	External RAM	8MB (EBS-30) N/A (EE200) N/A (EE201)	
	Clock frequency	100 MHz (EBS-30) 32 MHz (EE200)	
Display (EE200/1)	Active area	W 45 x H 22 mm	
	Number of dots	W 200 x H 96	
	Model	EG020AS183	
	Gray scale level	1 bit	Black and white
	Contrast ratio	10:1	
	Pixel pitch	111 dpi	
Connector (EBS-30)	Ethernet	10/100 Mbit	
Radio	Frequency	2.4gHz	
	Protocol	IEEE 802.15.4	

1.7. Electrical Specifications

1.7.1. Electrical Characteristics

EE200 / EE201:

Parameter	Typical	Unit	Remarks
Operating voltage	2.8–3.2	V	2 x CR2450 Lithium battery
Operating current	7-33	mA	During RF and image refresh only
Sleep current	1	uA	EE200/1 is in sleep state >99,9%

EBS-30:

Parameter	Typical	Unit	Remarks
Operating voltage	6.0	V	Power adapter
Operating current	<300	mA	

1.7.2. Main Battery

The main batteries in the EE200/1 are 2 CR2450 Li-Mn batteries:

- Nominal capacity: 2x600 mAh

- Nominal voltage: 3V.

- Low voltage: Less than 2.8 V

- Operating time: >8 years (default settings; 4 refreshes/day; 15-25° C)

The operating time of the batteries is based on a realistic estimation of any of the following factors that influence the battery life:

Factor	Default values and estimated averages	Correlation * (Higher is better)	Influence
Polling for data & reporting settings	Interval: 20 seconds Retransmissions: 10% Report ratio: 1 to 39	Yes No Yes	61%
Display updates	4 per day Temperature: 15-25° C	No No	20%
Out of range / base station OFF %	Not connected: 2% Number of channels: 5	No No	4%
Sleep time	99,9%	Yes	9%
Battery self-discharge	1 % / year	No	6%

* The correlation shows whether increasing the listed value increases or decreases the power consumption of the ESLs.

For example:

- Increasing the poll interval lowers the power consumption

- Turning off the base stations increases the power consumption

- A lower operating temperature increases the power consumption

- More display updates per day increase the power consumption

1.8. Serial Number

1.8.1. EBS-30 / EBS-31

The serial number can be found on the back of the Base station. The EBS-30 and EBS-31 also have a unique MAC-address which is used for both Ethernet and radio communication.

1.8.2. EE200 / EE201

The serial number of the ESLs is based on the 16 hexadecimal digits MAC-address of the RF-module.

However, the first 6 digits of the MAC address have been replaced by a letter.

- 'A' (instead of '00124B') for the EE200
- 'B' (instead of '3889DC') for the EE201

The default barcode that is shown on the display of the ESL is identical to the bar code on the back of the ESL.

1.9. Safety

Handle this product carefully. Do not deliberately subject it to any of the following.

1.9.1. Shock

Do not throw or drop the devices

1.9.2. Temperature Conditions

Do not use the devices at temperatures outside the specified range.

Do not pour water on the devices.

Do not throw the devices into the fire.

Do not forcibly bend the cable at low temperatures.

1.9.3. Foreign Materials

Do not immerse the devices in liquids.

Do not subject the devices to chemicals.

1.9.4. Other

Do not plug/unplug the connectors before disconnecting the power.

Do not disassemble this product.

The device may be damaged by power surges

2. Solution overview

5 different solutions are currently supported by the ESL server application:

- **CSV-file solution**. The CSV-file based solution operates by running the provided ESL server application that monitors an input folder in which CSV-files, containing the product and link databases, are being placed by a back office system.

- **SQL solution**. The SQL-based solution (MS-SQL, MySQL or Oracle) provides the possibility to monitor and control multiple stores using one central of multiple local SQL databases in combination with the provided web application. The local ESL server application monitors an SQL database using staging tables and processes them and inserts status information of all ESLs into the SQL database.

- **CSV / SQL hybrid solution**. Uses the provided ESL server application that monitors an input folder in which CSV-files, containing the product and link databases, are being placed by a back office system, but inserts status information of all ESLs into a local SQL database to allow remote monitoring of the ESL system.

- **External content management solution**. Allows the connecting of an existing content management system to the ESL server. Pre-generated images are placed into an input folder, which are automatically processed and send the corresponding ESLs. Logging of status information can be done by SQL if desired.

- **SQL API**. The ESL server supports a full SQL API, which can be used to create your own (web) application by executing SQL queries to (remotely) control and monitor the ESL system, while running the ESL server as background service.
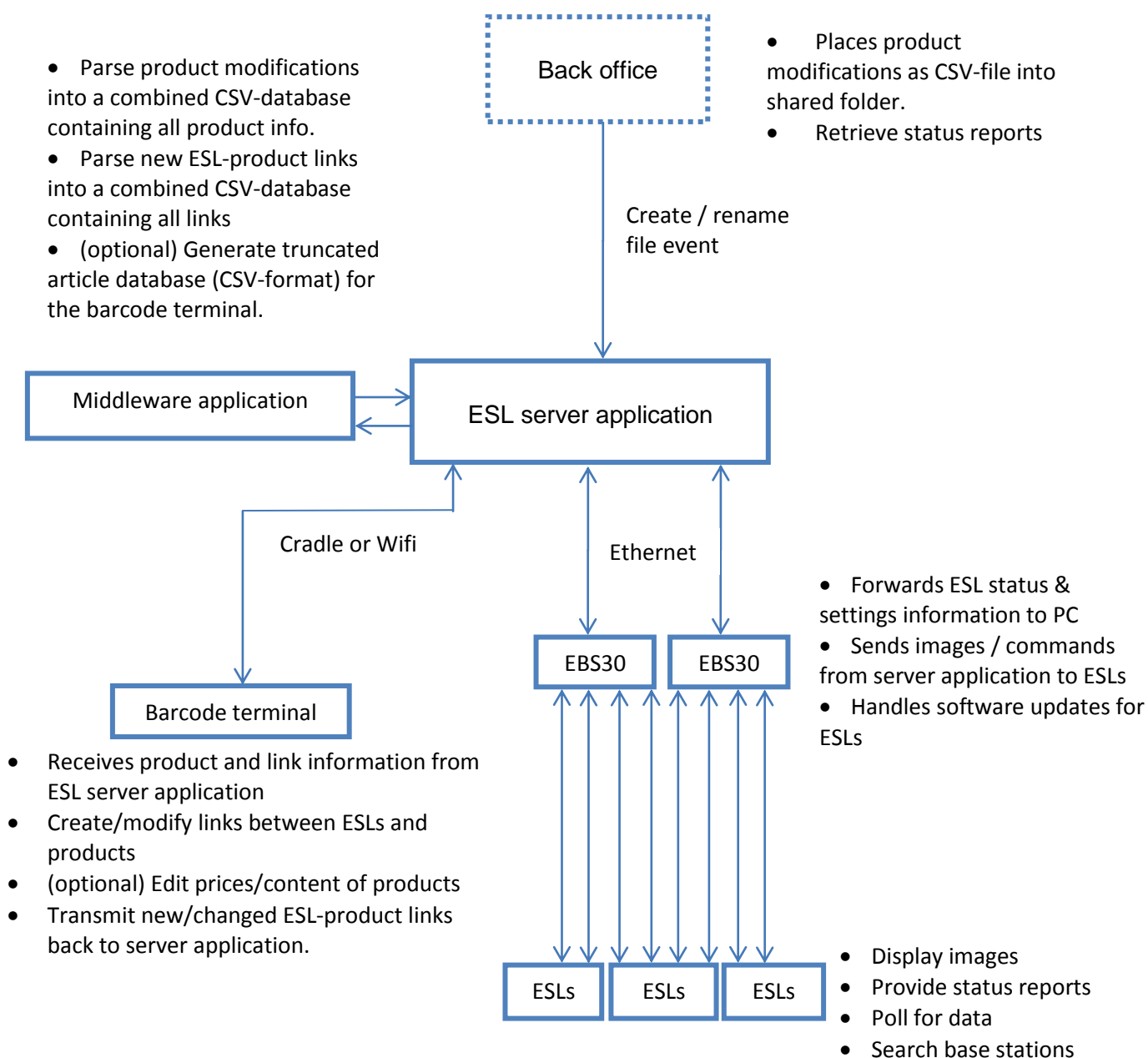
2.1.   CSV-file based solution

The CSV-file based solution operates by running the provided ESL server application that:
- Monitors an input folder in which product databases are being placed by a back office system and processes them.
- Can control, monitor and configure ESLs and base stations
- Allows manual editing of product and link data bases
- Creates/changes image templates, generates the ESL images and sent them to the ESLs

A global overview of this solution is shown below. The following chapters will describe the different parts of this system in more detail and how to get started.

- Parse product modifications into a combined CSV-database containing all product info.
- Parse new ESL-product links into a combined CSV-database containing all links
- (optional) Generate truncated article database (CSV-format) for the barcode terminal.

Back office

- Places product modifications as CSV-file into shared folder.
- Retrieve status reports

Create / rename file event

Middleware application

ESL server application

Cradle or Wifi

Ethernet

- Forwards ESL status & settings information to PC
- Sends images / commands from server application to ESLs
- Handles software updates for ESLs

EBS30   EBS30

Barcode terminal

- Receives product and link information from ESL server application
- Create/modify links between ESLs and products
- (optional) Edit prices/content of products
- Transmit new/changed ESL-product links back to server application.

ESLs   ESLs   ESLs

- Display images
- Provide status reports
- Poll for data
- Search base stations

2.2.   SQL Solution

The SQL-based solution provides the possibility to monitor and control multiple stores using one central of multiple local SQL databases in combination with the provided web application.

The local ESL server application(s) will:
- Monitor a local or central SQL database using staging tables and processes them.
- Inserts status information of all ESLs into the SQL database
- Controls, monitors and configures ESLs and base stations
- Allows manual editing of product and link data bases (optional)
- Creates/changes image templates, generates the ESL images and sent them to the ESLs

The web application can be added to provide remote and in-store control over the ESL system.

A global overview of this solution is shown below. The following chapters will describe the different parts of this system in more detail and how to get started.

- Create/modify links between ESLs and products
- View product information
- Edit prices/content of products (optional)
- Monitor ESL system status
- User Access Control

- Places product modifications in the SQL database using staging tables
- Retrieve status information from the SQL database and/or web application

Remote

Local

- Monitors the SQL database for modifications in the staging stables
- Inserts status information into the SQL database

Wifi

Ethernet

- In store access to the web application to create/modify links, view system status, product info and monitor ESL system status

- Forwards ESL status & settings information to PC
- Sends images / commands from server application to ESLs
- Handles software updates for ESLs

- Display images
- Provide status reports
- Poll for data
- Search base stations

14

2.3. CSV / SQL hybrid solution

The CSV / SQL hybrid solution matches the CSV-solution with as only difference that all status information on the ESLs and base station on logged into a local SQL database which can be retrieved by the back office system. See chapter 4.3 for more information.

2.4. External content management solution

When using external content management solution the back office or content management system handles the generation of the images and the linking of products to ESLs.

The ESL server application basically only monitors an input folder for new images and distributes them to the corresponding base stations / ESLs.

Status information will be logged into a local SQL database. See chapter 4.4 for more information.

2.5. SQL API

(My)SQL can be used as API to create your own (web) application by executing SQL queries to control and monitor the ESL system, while running the ESL server as background service.

See chapter 4.2 for information on how to use the SQL product and link staging tables to link and update ESLs and use the status tables to monitor the ESL system.

See chapter 9 on how run the ESL server as service.

See 'Appendix E: Using the SQL API' for more information on this API.

For maximum flexibility the SQL API can also be used in combination with the CSV/SQL hybrid solution and with the external content management solution.

3.   Getting Started

3.1.   Setting up the hardware and software

This SDK contains the following pieces of hardware and software that will have to be installed in order to get an ESL demonstration or pilot up and running:

- The ESL Demo kit installer containing:
  o The ESL server application
  o The ESL Web application
  o Example database files and image templates
  o Firmware for the barcode terminal, base station and ESLs
  o Appload (optional)
  o USB drivers for the barcode terminal cradle (optional)
  o User manual
- 1 EBS30 Base station(s) that includes:
  o 6V Power supply
  o Ethernet cable
- 6 EE200 or EE201 2-inch e-paper ESLs (batteries included)

If you want to be able to quickly link ESLs to products, you can use either an:

- OPH1004, H13, OPN2002 or OPL9728 barcode terminal (incl. communication cradle/cable)

    Or

- H21, H22 or H32 using Wifi (requires a local web server running the provided web application)

3.1.1. Installing the ESL Server application

To install and run the software you will also need a PC running Windows 7 or Windows XP (with server pack 3) as server.

Before installing the software, make sure you have **administrator rights** and the **PC is fully updated with al Windows updates**, including the latest Internet Explorer updates.

If your PC doesn't have .NET Framework 2.0 (or higher) installed, you will have to install it prior to installing the ESL Server application.

Run the SDK installer by starting the 'setup.exe' file.

Select the components that you wish to install. If you haven't installed Appload or the Opticon USB drivers on your PC before, make sure you install those components as well.

3.1.2. Installing and finding the EBS30 or EBS31 base stations

The base station can easily be installed by powering it up, connecting it to the Ethernet with DHCP support and placing it within 10 meter (the actual maximum range is bigger) of your ESLs. The PC will need to be connected to the same LAN, so it can communicate with the base station*.

After starting the application for the first time, no base station will be available to communicate with ESLs.

Press the 'Find' button on the 'Base stations-tab to find the connected base station.
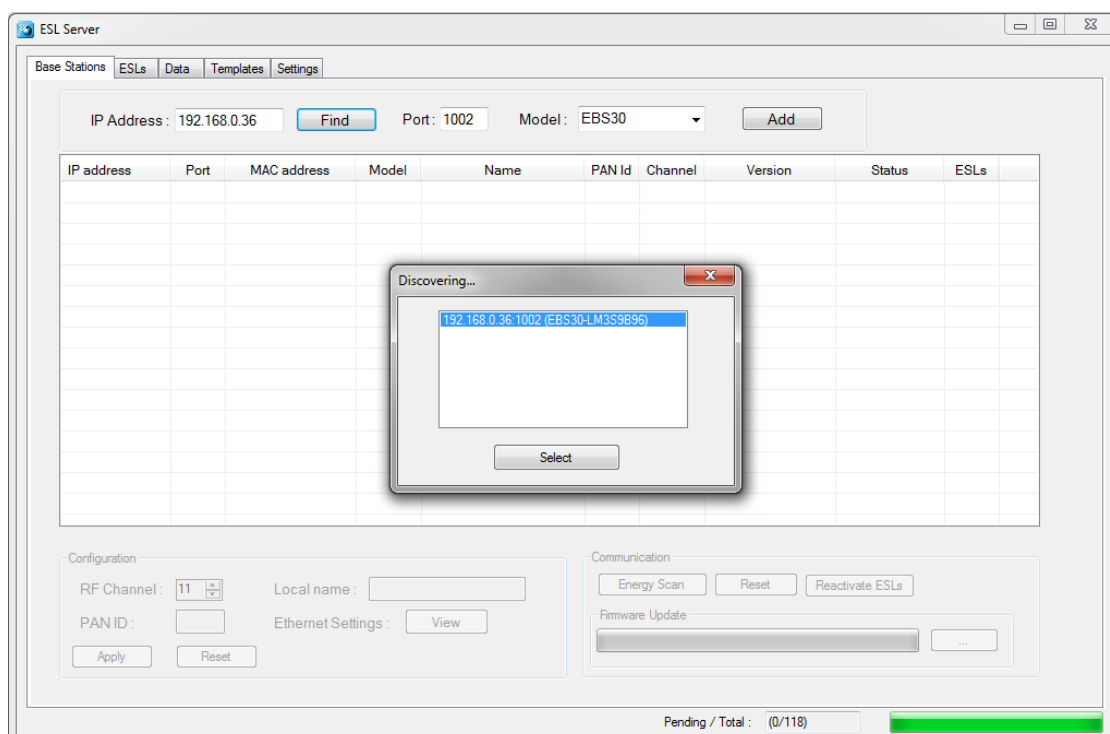
Select the discovered base station and then press the 'Add' button to connect your PC with the base station.*

If activated ESLs are nearby, then the ESL count will start incrementing as they start connecting to the base station. However if the ESLs are still 'deactivated', they will first have to be reactivated.

Note 1: when the base station is connected to the LAN with DHCP the IP address can vary over time, making it necessary to remove the old base station entry by pressing the 'Delete'-key of your keyboard and discovering the base station again using the 'find' button. Using static IP leases in the DHCP server can resolve this problem.

*Note 2: when the base station is connected to the LAN without DHCP, the default IP address/subnet mask of the base station is **169.254.19.63 / 255.255.0.0**. Configure a PC or laptop on the same subnet to change this using a web browser.*

3.1.3. Installing and activating the ESLs

On arrival the ESLs usually already have their two CR2450 batteries inserted.

However, to allow safe transport by airplane, it is likely that the ESLs will show a 'DEACTIVATED' message in their left top corner. This means the ESLs do not emit any radio signals and won't be searching for base stations at this point.
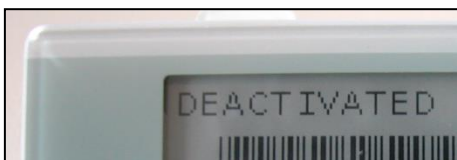
To reactivate these ESLs place them nearby the connected base station and select the base station in the list view of the 'Base stations'-tab. This allows you to press the 'Reactivate ESLs' button in the bottom right corner.

After pressing this button, a dialog will appear showing a progress bar and how many ESLs have been reactivated so far.

Since the ESLs will only wake-up for a split second every few minutes, it can take more than 15 minutes before all ESLs have been successfully reactivated. You can verify this by checking whether the 'DEACTIVATED' message has disappeared from the screen.

Active ESL's will be periodically seek for an Opticon base station, so as soon as an Opticon base station is within range, they will associate with it and start sending data requests and status information to it. When a base station is turned off or an ESL is moved out of range, an ESL will time out after a couple of minutes and starts seeking for another base station in range. See chapter 3.3 for more information about configuring ESLs.

It might be possible that even though the ESLs are working properly, they still don't connect the base station. This might be caused by (Wifi) interference on the default RF-channel of the base station. To resolve this problem, see chapter 3.2.1

3.1.4. Mounting ESLs

Since there are many types of mounting clips to mount shelf labels to shelves, the ESLs of the demo kit are supplied without mounting clips. Please contact Opticon for more information on the mounting possibilities of our ESLs to your shelves. For simple demonstration purposes with small amounts of labels one might also consider using double sided foam tape as a temporarily solution.

3.2. Configuring the base stations

Base stations have very few configurations besides the configuration of the Ethernet settings (which will be described in chapter 3.2.4). The only vital configuration is the RF channel on which the base station operates.

### 3.2.1. Configuring the RF channel

The base station operates on a single RF channel, which can be selected in a range of channel 11 to 26 (as specified by the IEEE 802.15.4 standard).

When using multiple base stations it's strongly recommended to use a different RF channel for each base station within range to improve the maximum throughput of data and reduce data collisions.

Since the base station / ESLs, which apply the IEEE 802.15.4 standard, operate on the same frequency band as i.e. WIFI and Bluetooth it's important to make sure that a base station does not operate on the same RF channel as any nearby WIFI network to minimize interference.

To detect which RF-channels are least active, the base station can perform an energy scan on all 16 available channels and graphically show the energy levels in a real-time graph.

To check the energy levels on the RF bandwidth select the base station in the 'Base stations'-tab and press the 'Scan' button.
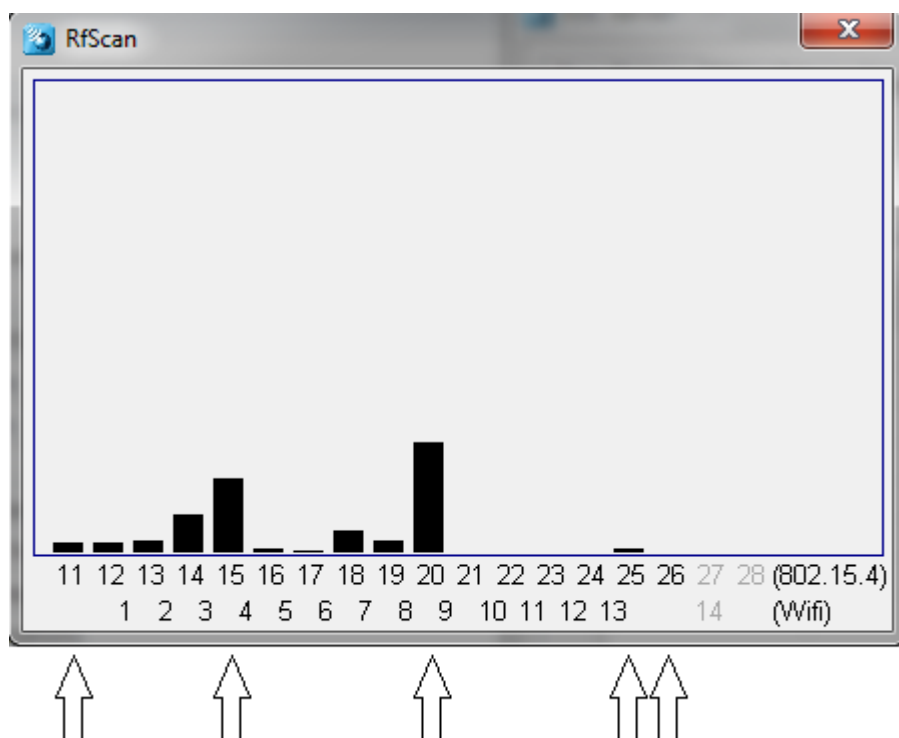
Since the EE200/1 ESLs will (by default) only look for base stations on **channels 11, 15, 20, 25 &
26**. Look for the RF channel with the lowest energy level (on average) of these 5 channels.

If the RF channel of the base station is set to a channel outside this selection, the ESLs will not
find the base station. If you do wish to use any of the other channels, then it's necessary to first
configure a different RF channel selection into the ESLs before configuring the base stations.
(See chapter 3.3.6)

*Note: If an ESL is not associating to any base station and its RF channel range is unknown, then
configure a base station to **channel 11**. This channel will always be scanned by an ESL to make
sure they can always be found and reconfigured without having to look for it on all channels.*

After determining the least active RF-channel, close the RfScan-dialog. After that you can change
the 'RF Channel' in the 'Configuration' section of the 'Base stations'-tab and press 'Apply' to
confirm.

After the RF channel is changed the base station will automatically reset to apply the new setting.

Be aware that any ESLs that were already associated with the base station before changing the
RF channel will lose their connection with the base station. By default it will take 30 polls (times

the poll interval of 20 seconds) before the ESLs will time out and start looking for another base station.

To avoid having to wait a couple of minutes before the ESLs starts looking for the base station on the new RF-channel it's possible to disassociate all the ESLs from the base station prior to changing the RF channel. To do so, select all ESLs in the 'ESLs'-tab that need to be disassociated, then select the 'Kick from base station' action and press the 'Start' button. All ESLs should now start disassociating from base station. Once they're all gone, quickly change the RF channel and the ESLs should start to reappear again on the new channel.

3.2.2. Changing the local name

To make it easier to distinguish multiple base stations, it's possible to assign a name to base stations. This can be done by editing the text field besides 'Local name' and pressing the 'Apply' button afterwards to store the name inside the base station.

3.2.3. Changing the PAN ID / LAN ID

This feature requires EBS3x firmware version: IBMV0015 or higher
and EExxx firmware version IBxV0058 or higher!

Base stations are addressed by the ESLs by their PAN ID (2 bytes) and a 16-bit LAN-ID.

The **PAN-ID** makes it possible to have multiple base stations operating on the same RF channel. By default the PAN ID is the same as the 2 least significant bytes of the MAC-address. In the extremely rare case that you will have two base stations that have the same default PAN ID, it's necessary to change the PAN ID manually. In this case, edit the PAN-ID to a unique 4 digit (hexadecimal) value and press the 'Apply' button afterwards to store the name inside the base station.

The **LAN-ID** makes it possible to run two or more parallel ESL networks next to each other. By default the LAN-ID is 0000, which means any Opticon ESL will connect to this base station. To run two ESL networks in parallel, change the LAN-ID of both networks to non-zero.

Do this by configuring the LAN-ID of all base stations of both networks. Then make sure the LAN-ID of all ESLs is cleared (=default), by using the ESL action: 'Clear LAN Id'). Then move all ESLs to a base station of the desired LAN-ID. Once all ESLs are associated to the correct network, choose the ESL action 'Set LAN Id' to lock the LAN-Id of the ESLs.

Be aware that any ESLs that were already associated with the base station before changing the PAN-ID or LAN-ID will lose their connection with the base station. By default it will take 10 or 20 polls (times the poll interval of 20 seconds) before the ESLs will time out and start looking for base stations again.

3.2.4. Configuring the Ethernet settings

By default the base station is configured with DHCP enabled. If DHCP is not available, then the EBS-3x will be available on the following IP address and subnet mask: **169.254.19.63 / 255.255.0.0.**

If DHCP is used, make sure the DHCP server always leases the same IP to the same base station to prevent having to find and add the base station again every time that the IP-addresses changes.

To change the Ethernet settings, you first have to find and add the base station in the 'Base stations'-tab of the ESL server application. After that, you can open the embedded webpage of the EBS-3x with your default web browser by pressing the 'View' button next to 'Ethernet settings'.

It's also possible to access this webpage directly from your web browser using the IP-address of the base station.

After that you can change all the settings of the base station by going to the 'Configure' page.

Important notes:

- *If DHCP is disabled and the current IP address and subnet mask are unknown, then it's possible to recover the EBS-3x by applying power the device with the reset button pressed. This will re-enable DHCP temporarily to allow reconfiguration of the EBS-3x on a network with DHCP enabled.*

- *If DHCP isn't available or working, then it's possible to connect the EBS3x directly to the Ethernet port of your PC with the following IPv4 set-up: 'Obtain IP address automatically disabled; IP: 169.254.19.61; Subnet: 255.255.0.0. After that the EBS-30 can be discovered and configured on the default IP-address: 169.254.19.63 using the configuration webpage.*

- *Do not use a WiFi connection to connect your ESL server to the EBS-3x's to unsure a reliable connection.*

- The default password of the configuration page is 'admin'.

- *The base station can also be controlled over the Internet or via a public network, but it won't be a secure solution, since SSL isn't supported yet in this SDK nor is the webpage password protected.*

3.2.5. Factory default

To set the EBS-3x back to factory default, apply power with the reset button pressed and keep it pressed for 15 seconds. This will re-enable DHCP and set the IP address and subnet mask back to the default values: 169.254.19.63 / 255.255.0.0. Also the other configurations will reset to factory default.

Important: If DHCP is disabled and the current IP address and subnet mask are unknown, then it's possible to recover the EBS-3x by applying power the device with the reset-key pressed. This will re-enable DHCP temporarily to allow reconfiguration of the EBS-3x on a network with DHCP enabled.

3.3. Configuring of ESLs

There are a number of configurations that can be made in the ESLs to control, customize them and to regulate power consumption.

To edit configurations of an ESL, go to the 'ESL'-tab base station tab and select the ESLs that you which to configure. To select multiple ESLs, use the shift- or ctrl-key.
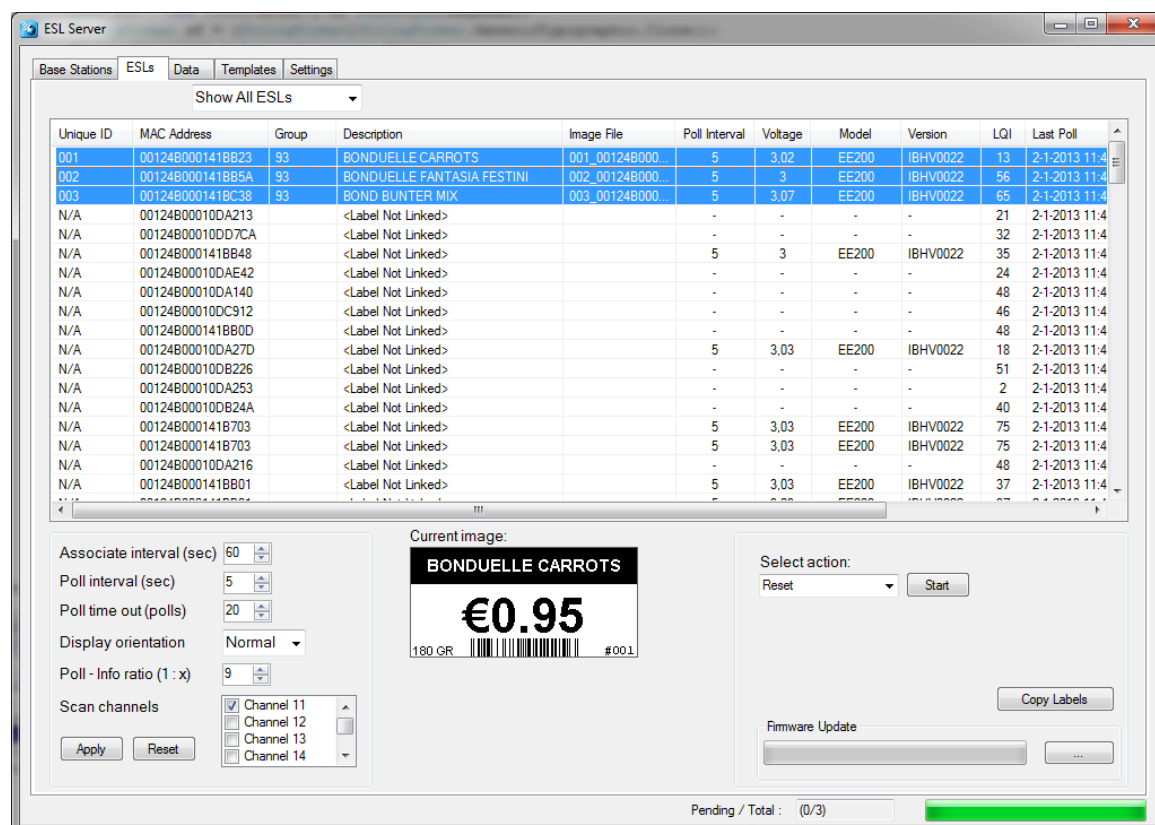
After doing so, the current settings of the first selected ESL will be shown below.

Note: If the base station has not yet received the settings from this ESL, then you'll have to wait till the settings have been received from the ESLs. These are automatically requested the next time the ESL polls for data.

There are 6 configurations that can be edited. These configurations will be described in the following chapters:

- Associate Interval
- Poll interval
- Poll timeout
- Display orientation
- Poll - Info ratio
- Scan channels
- Temperature range

3.3.1. Associate interval

When an ESL is not connected to any base station it will automatically start looking for nearby base stations. Since searching for base station consumes a relative large amount of power it will not continuously search, but only once every couple of seconds. The associate interval determines the time (in seconds) between two associating attempts of an ESL. The default associate interval is currently 60 seconds.

Please note that ESLs consume much more power while not being connected to a base station than when they're connected, so it's strongly recommended to never turn off the base stations once a network of ESL's is operating. If ESLs are not being used for a long period of time the associate interval can best be set to a much longer interval like 300 seconds. For an even longer period of time, removing the batteries might be the best option.

The amount of power that is being consumed by the ESL, when it's not connected to any base station is also strongly related to the number of RF channels it is configured to search for base stations. More information about the configuration of the 'Scan Channels' can be found in chapter 3.3.6.

3.3.2. Poll Interval

The poll interval specifies the interval (in seconds) at which the ESL sends a poll for data to the base station. This parameter is the most significant configuration of an ESL since it strongly determines the response time of an ESL (how fast is an image displayed) as well as the battery life time.

By default the ESLs poll every 20 seconds once they're connected to a base station. This means that to response time of an ESL can be anywhere in between 0 to 20 seconds (unless the poll doesn't arrive at the base station due to interference).

The poll interval greatly influence the battery life time, so lowering it is not recommended. For example, changing the poll interval from 20 seconds to 5 seconds will roughly triple the power consumption and thereby shorten the battery life time by a factor of 3.

An advantage of making the poll interval bigger is the reduction of the used bandwidth per ESL, making it possible to connect more ESLs to a single base station.

3.3.3. Poll timeout

The poll timeout specifies the number of polls without acknowledge from the base station after which the ESL will time out its connection and start searching for another base station. The default poll time out is currently 30 polls, so by default the time out is 30x15 seconds = 450 seconds.

3.3.4. Display orientation

It is allowed to mount ESLs up-side-down. To make sure that the images are displayed correctly, the display orientation can be changed to 'Flipped'.

3.3.5. Poll - Info Ratio

The poll info ratio determines how often ESLs will automatically send their current status and settings to the base stations. By default the ESLs will transmit this information once every 40 polls, so a ratio of 1:39.

The status and information of an ESL is also requested immediately by selected un ESL in the ESL-tab. This will cause the base station to request the settings the next time this ESL polls for data.

The poll info ratio is set to a relative high value, because settings and status usually don't vary a lot. Also sending settings and status information uses more bandwidth and consumes more power than regular polls for data.

3.3.6. Scan channels

Once an ESL has associated (connected) to a base station, it operates on a single RF channel. However when it's not yet connected to a base station, the ESL will automatically scan a range of RF-channels for base stations. The range of RF channels that an ESL will scan for base station is configurable. All channels in the range of channel 11 to 26 can be scanned by adding the channel in the dash-separated list of channels.

By default the channels 11, 15, 20, 25, 26 are enabled. Channel 11 can't be disabled to make sure that when an ESL that is not associating to any base station and its RF channel range is unknown, it can always be retrieved by configuring a nearby base station to channel 11.

3.3.7. Temperature range

Configures the temperature range at which the ESL is installed. Because e-paper displays become slower at low temperature, lowering the temperature range compensates for this to avoid the display not updating correctly.
**Only use this feature when the display is not updating correctly under low temperatures,** because lowering the temperature range increases power consumption, because it takes longer to update the display. Also be aware that low temperatures decrease the battery capacity and thereby also the battery life time.

3.4. Monitoring of ESLs

All available information of an ESL can be displayed in the ESL-tab. Each column can be enabled or disabled by right-clicking of the column-header and (un)selecting items of the list below.

- Unique ID   Unique ID of the product to which the ESL is linked.
- MAC   16-digit MAC address of the ESL
- Group   Product group as configured in the 'Data'-tab.
- Description   Product description as configured in the 'Data'-tab.
- Image file   File name of the generated image for this ESL
  - o   <Empty>   ESL is not linked to any product
  - o   Light green   Image on ESL is up-to-date

- o **Dark green** Image is currently updating
- o **Yellow** Image is currently updating (previous attempt failed)
- o Orange Image is ready for updating, but the ESL has not polled since
- o **Red** Error: Image file could not be found / Image update failed
- o Light grey ESL is missing / not associated to any base station
- Poll Interval Poll interval (see chapter 3.3.2)
- Poll Timeout Poll time out (see chapter 3.3.3)
- Associate Interval Associate interval (see chapter 3.3.1)
- Battery voltage Current battery voltage (deprecated)
- Model Hardware model of the ESL (i.e. EE200 / EE201)
- Version Software version of the ESL firmware
- SubVersion Software version of the sub-processor of the ESL (if present)
- Image Id Image Id of the image on the ESL. (0=default image; 1=normal image)
- Backlight Backlight setting of the ESL (only for TFT based ESLs)
- Display Orientation Display Orientation (Normal or Flipped; See chapter 3.3.4)
- LQI Indicator of the signal strength from an ESL (measured by base station)
  - o Dark orange (1-3) Link quality is poor and high packet loss (>60%)
  - o Orange (4-7) Link quality is poor and significant packet loss (>20%)
  - o **Yellow** (8-10) Link quality is poor, but no significant packet loss (<20%)
  - o Light green (>10) Link quality is good and no significant packet loss
- LQI (Rx) Indicator of the signal strength from an ESL (measured by ESL)
- Last Poll Time of the latest poll that has been received from the ESL
- Base Station IP-address of the base station to which the ESL is currently connected
  - o Light green Base station is connected
  - o Orange Base station is not connected
  - o Light grey ESL is missing / not associated to any base station
- Status Current state of the ESL.
- Temperature range Temperature setting: 0=default (higher the value, the lower the temp.)
- Image status Shows the current status of the image update
- Firmware status Shows the current status of the firmware update
- LAN ID Shows the preferred LAN-ID on which the ESL will look for base stations (0000=associate to any base station, XXXX = only associate to base stations with same LAN-ID or with 0000 only)
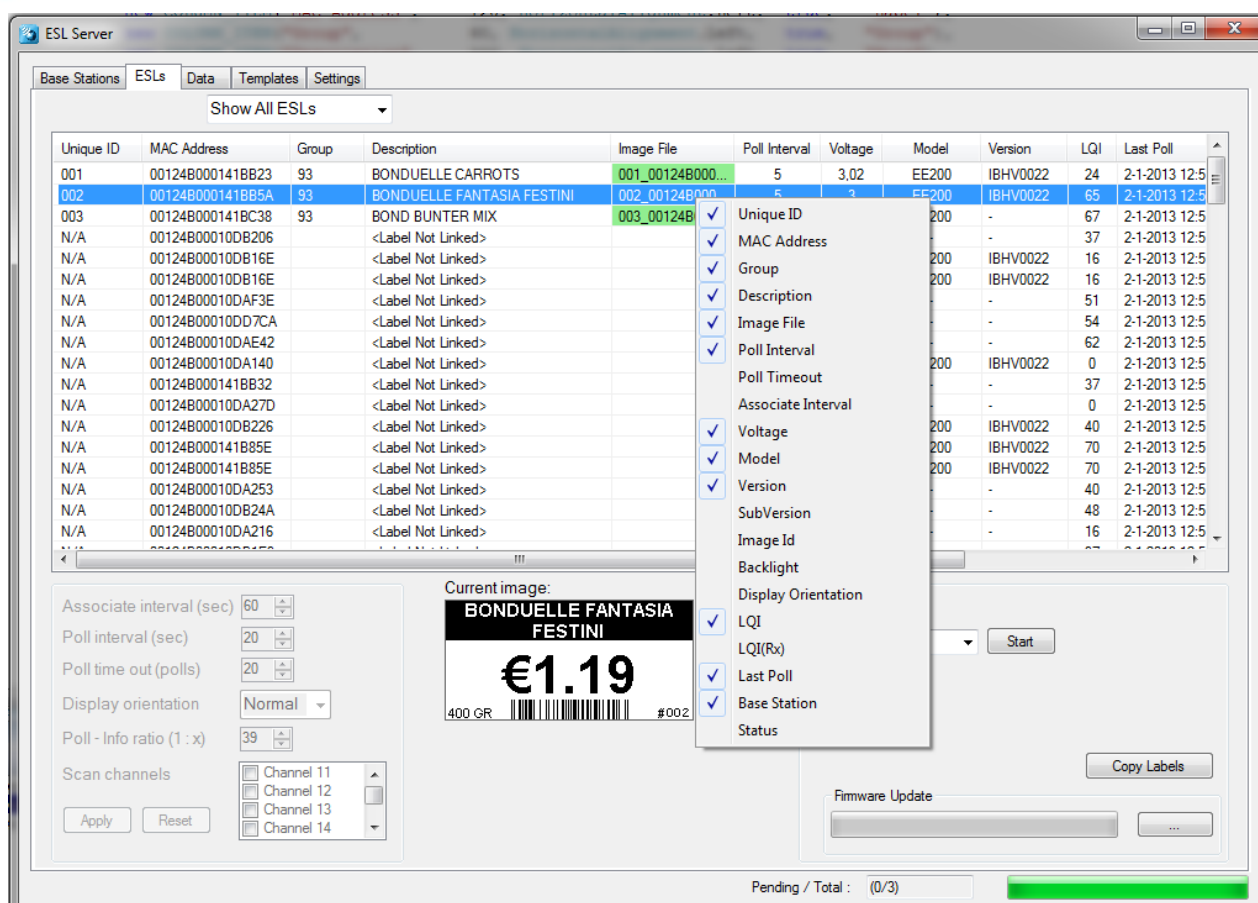
3.5.   Filtering ESLs

Using the newly added filter options in the ESL-tab it's possible to sort and filter on almost any ESL property.
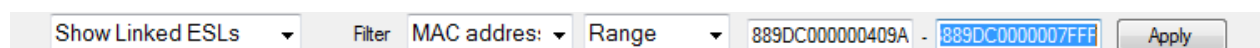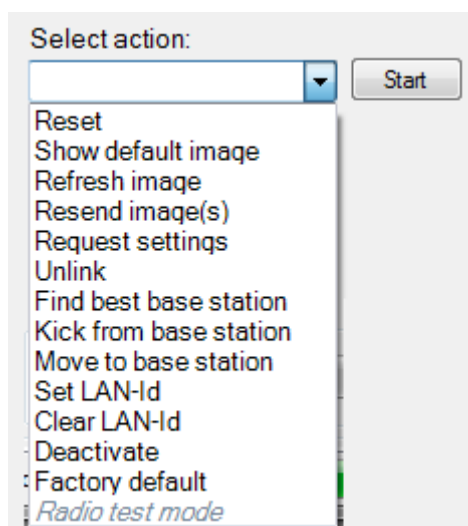

3.6.   Controlling ESLs (Actions)

ESLs can be further controlled by using the 'Actions' of the ESL-tab. To use these features, select one of more ESLs and then select the action that you wish to execute.

Features that are unavailable for this ESL will be grayed out. Below a description of the different actions is given.

3.6.1. Actions

- Reset   Resets the firmware of the ESL, causing it to reassociate with the same base station
- Show default image   Make the ESL show the default image with its MAC address (if the ESL is linked, the default image is shown for 15 minutes after which the product image is send again)
- Refresh image   Refreshes the current image on the ESL
- Resend image   Resends the current image to the ESL
- Request settings   Force the ESL to send its current settings
- Unlink   Unlinks the ESL from the link database
- Find best base station   Search for a base station with a better link quality
- Kick from base station   Search for another base station
- Move to base station   Move ESL to the specified base station
- Set LAN-Id   Sets the LAN-Id (see chapter 3.2.3)
- Clear LAN-Id   Clears the LAN-Id (see chapter 3.2.3)
- Deactivate   De-activate the ESL for transport or storage
- Factory default   Resets the firmware of the ESL to default
- Radio test mode   Used for certification only (disabled by default)

3.6.2. Load balancing

ESL server 1.45 and higher has to possibility to automatically and manually re-balance the ESLs over all available base stations. Enabling this option on a daily basis improved the average link quality of the ESLs, because all ESLs with poor link quality will check if there's a better base station available.
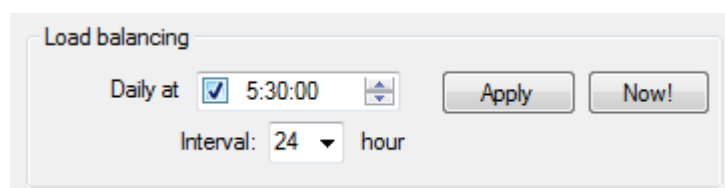
Especially after a new base station was added or a base station has been temporarily offline, it can be beneficial to execute load balancing manually.

Please note that load balancing significantly improves if ESL's run firmware IBxV0058 or higher.

Be aware that during the rebalancing of the ESLs, they can appear to be offline for 1 to 15 minutes, so it's safest to execute the daily load balancing at night to avoid slowing down the image updating process at daytime.

3.6.3.   Firmware update

To upgrade the firmware of ESLs, select one of more ESLs and then press the 'Firmware update' browse button and select the correct firmware file for these ESLs, which can be found in the 'Firmware' sub-folder of the installation. Make sure the first 3 characters of the firmware file match the reported first 3 characters of the reported firmware version by these ESLs.
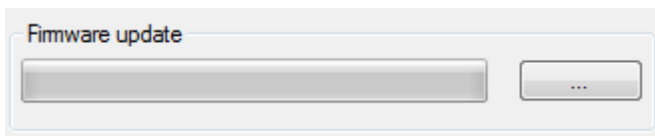
3.7. Installing the bar code terminal

The quickest method to link a large amount of ESLs to products in your database is by using a bar code terminal. This is done by scanning the MAC address bar code on the ESL label followed by scanning (or entering) the unique ID of the product. The unique ID can be anything: an EAN code, article number, location, room number, etc. As long as it's a unique identifier it can be linked to an ESL.

Opticon provide software for a wide variety of barcode terminals to perform this task. The following Opticon bar code terminals are currently supported:

- OPN2002 (bar codes only; uses a mini USB cable)

- OPH1004 (bar codes & keypad; requires a CRD1001 cradle)

- H13 (bar codes & keypad; requires a CRD13 cradle)

- OPL9728 (bar codes & keypad; requires a CRD9723RU cradle)

- H21 (bar codes & keypad; requires Wifi and the provided web application)

- H22 (bar codes & keypad; using Wifi and the provided web application)

- H32 (bar codes & keypad; using Wifi and the provided web application)

More information on how to set-up the barcode terminal can be found in chapter 7.

4. Database configuration

The ESL server support 3 different database formats:

<u>**CSV**</u>

Comma separated files. It's easy to setup and it's suitable for smaller projects and demos with a limited number of ESLs. Disadvantage is that the ESL server is difficult to control and monitor by an external server due to the limited feedback on the status of the ESL system. Also CSV-based systems tend to become slow when used with large numbers (thousands) of ESLs. See chapter 4.1.

<u>**MySQL**</u>

MySQL is an OpenSource relational database system. Requires knowledge about SQL databases and MySQL has to be installed on your system. MySQL provides the possibility to monitor and control multiple stores using one central of multiple local SQL databases in combination with the provided web application. MySQL is suitable for large number of ESLs per ESL server. See chapter 4.2.

<u>**MS-SQL**</u>

MS-SQL is a relational database system from Microsoft. Same advantages and disadvantages as MySQL and Oracle. MS-SQL has a slightly better performance over MySQL, because it the ESL server is developed on and runs on a Microsoft platform. A disadvantage is that it's harder to control an MS-SQL database using the provided web-based application. See chapter 4.2.

<u>**Oracle**</u>

Oracle is a relational database system from Oracle. Same advantages and disadvantages as MySQL and MS-SQL. Oracle databases have tons of features, but this also makes them more complex to set-up and requires serious knowledge about this database format. Currently the provided web-based application does not support Oracle. See chapter 4.2.

<u>**CSV & SQL hybrid**</u>

It is possible to combine the advantages of CSV and SQL by configuring a hybrid configuration with both CSV (to provide the product databases and ESL-product links) and SQL (to log all ESL and Base station status information). This makes it easy to link the back-office to the ESL server using CSV-files and all status information will be available for remote monitoring. See chapter 4.3

4.1. CSV Data base configuration

When using CSV-files, the ESL server application is triggered by files that are being placed in a specified 'Input' folder by the back office. These can be either CSV, image or template files. Using a shared folder it's possible to allow external access by the back office from anywhere in the world.

If the back office system is not yet available, then it's also possible to edit prices, links and products manually in the ESL server application in the 'Data'-tab.

Alternatively there's also the possibility to manually place (and rename) some CSV-files in the 'Input folder' (see chapter 4.1.1) for demonstration purposes and testing.

To select the CSV-database format, select 'CSV' in the 'Data'-tab.
See chapter 6 for information about on setting up the back office system.

4.1.1. Changing the database format and settings

Important: Whenever you change the database format (i.e. by changing the field sequence or by adding/removing fields) or change the database set-up, then it will be necessary to place a new full copy of the product database in the 'Input'-folder. This will automatically trigger the parsing of database with the new format or settings.

(a)   **Input Folder**

Directory in which the back office uploads the new product data files (see chapter 6.1. The database files must have the file extension **.csv** in order to be detected by the ESL server application.
Besides database files, it's also possible to place ESL-product link files (see chapter 6.2), template files, images and template files in this folder for processing.

(b)   **Output Folder**

Output folder for the middleware application. It will contain 2 CSV-files:
*   dbase.csv (combined product database)
*   linked.csv (ESL-products links)

When using a bar code terminal for linking, it can also contain the following file:
*   scan.csv (truncated article database)

Any processed CSV-files from the Input folder will be moved to the sub-folder:
*   /Processed

(c)   **CSV parser (optional)**

Important note: The CSV parser field can best be left empty if no discount percentage is used in your CSV-file!

Executable of the middleware application that parses new product databases and product-ESL links into a combined database (dbase.csv) and a combined product-ESL link database (linked.csv). When a bar code terminal is used, it also generates a truncated article database (scan.csv) that is sent to the barcode terminal to allow linking of products to ESLs.

By default the CSV-parser expects the following field sequence in the CSV-files:

```
<Modification code>;<Unique ID>;<Barcode>;<Description>;<Group>;<From price>;<Sell
price>;<Discount %>;…;etc.
```

If the field sequence of the supplied database is different, then this can be changed by altering the following field numbers.

Important notes:
- *The first field is field number '0', the second is '1', etc!*
- *The default CSV-parser uses the field 0 to detect whether it's a complete or modification database file, so please don't remove this field.*

Unique Id field: Field number of the unique product ID in the CSV-record. Even though the product ID must be unique, it is allowed to have multiple records with the same ID in your database with different barcodes.

Barcode field: Field number of the product barcode in the CSV-record. If the barcode field is the same as the Unique Id field or when the barcode field isn't unique or present, use the Unique Id field number for both configurations!

Description field: Field number of the product description in a CSV-record.
(The description is shown by the barcode terminal when a link is created, so the user can verify the correct product was linked).

Group field: Field number of the product group in a CSV-record. If no group field is present in the CSV-record, set the field number to '0'.

From price field: Field number that holds the 'from price' from the article. If no 'from price' is present in the CSV-record set it to the 'sell price field'

Sell price field: Field number that holds the current 'sell price' from the article. If no 'sell price' is present in the CSV-record set it to the '0'

Discount % field: Field number which holds the automatically calculated discount percentage. If the 'from price' and 'sell price' fields are equal or set to '0', then the discount percentage isn't calculated.

Currency Symbol: Holds the currency symbol. Used by the image generator to add the currency in front of the price. (Must be specified in the image template using %C%, see chapter 5)

Decimal Symbol: Holds the decimal symbol for prices, which can be either a dot or comma. If the user accidently types in a comma instead of a dot (or vice versa) in a price field, then the CSV-parser will automatically correct this.

Display leading zero: Specifies the price formatting when the price is smaller than one standard currency (i.e. 0.75). The CSV-parser can automatically format prices in any of the following three formats: 0.75, -.75 or .75.

4.2. SQL Data base configuration

The ESL server supports both MS-SQL, MySQL and Oracle, which need to be installed besides the ESL server application in order to use it.

- MS-SQL Express Server 2012 (with tools) is available at Microsoft.com.

- MySQL can be installed separately or using XAMPP (see chapter 8)

- Oracle Express can be downloaded from ORACLE.com

The ESL server application will generate most SQL tables automatically, except for the product tables, which contain the product database.

A demo query for MySQL can be found in 'Appendix B: MySQL example query and configuration'

A demo query for MS-SQL can be found in 'Appendix C: MS-SQL example query and configuration'

A demo query for Oracle can be found in 'Appendix C: MS-SQL example query and configuration'

Important note: make sure the SQL login, that is to be used by the ESL server, has sufficient rights to read, write and create these tables. Also makes sure that the used login doesn't require periodic password changes.

4.2.1. Tables

When using MySQL, MS-SQL or Oracle, the ESL server uses 10 SQL-tables in order to work properly, which have a couple of restrictions.

- A **product table** containing the product information, which allows the image content on the ESLs to be automatically generated.

- A **product staging table**. Used to serve as the interface to add, modify and removes products from the product table

- **Link table** This table is automatically generated and is used to containing the links between products and links

- **Link staging table**. This table is automatically generated and is used as the interface to create and remove links from the link table

- **LabelStatus table.** This table is automatically generated and is used to store all information about each ESL in the server. This table is described in chapter 4.2.6.

- **BaseStationStatus table**. This table is automatically generated and is used to store all information about each EBS3x base station in the server. This table is described in chapter 4.2.7.

- **Logging table**. This table is automatically generated and is used to store events and alerts and exceptions to help trouble shooting in case of problems.

- **Users table.** This table holds the users, encrypted passwords and access rights

- **Change log table.** This table is used to log user activity.

- **(Optional) Product look-up table** containing just bar codes and their related product ID. It's not actually used by the ESL server application, but it's used by the included Web-app to link products to ESLs by scanning the barcode of a product followed by the barcode on the ESL. This table is described in chapter 4.2.8.

The following chapters will describe these tables in more details.

(The ESL server also creates 3 'action' tables, which are created to allow remote control over the ESL server using SQL queries. See Appendix E: Using the SQL API.

4.2.2. Link table

Important note: Only the ESL server is allowed to modify this table directly. A link staging table is used for changing the content of this table, which is described in the following chapter.

The ESL server requires a link table that contains the links between the products and ESLs.

If no Link-Table is supplied, then one will be automatically created (called "links") in the following format:

```
CREATE TABLE links
(
[ID] VARCHAR(40),
[Variant] VARCHAR(1),
[MAC] VARCHAR(16) NOT NULL PRIMARY KEY
)
```

If you wish to name the table or columns differently, this is allowed. You can do so by creating a table with the same format, but different table and column names.

The 1st column (ID) should contain the unique product code.
The 2nd column (Variant) should contain the ESL software variant ('H'=EE200/201, 'I'=EE201)
The 3rd column (MAC) should contain the MAC (or IEEE) addresses of the ESLs.

It is allowed to link the same product to multiple ESLs.

4.2.3. Link staging table

Staging tables are the interface tables between the remote database from the back office and the local SQL databases used by the ESL server.

If no Link staging-Table is supplied, then one will be automatically created (called "Links_Staging") in the following format:

```
CREATE TABLE links_staging
(
[ID] VARCHAR(40),
[Variant] VARCHAR(1),
[MAC] VARCHAR(16) NOT NULL PRIMARY KEY,
[DELETE] VARCHAR(1)
)
```

Changed links should be inserted in the link staging table and the ESL server will automatically detect and process this data and insert the data into the Link table. If the processed data requires an ESL to be updated, it will automatically generate a new image and sent it to the ESL.

The link staging table must have the same formatting as the link table, but should also contain a 'DELETE' column. The 'DELETE' column makes it possible to remove a link from the link table, by placing an 'X' in this column.

4.2.4. Product table

The product table should at least contain all the product information, which should be shown on the automatically generated image of ESLs.

Example queries for a filled product (and product staging) tables can be found in:

- Appendix B: MySQL example query and configuration

- Appendix C: MS-SQL example query and configuration

- Appendix D: Oracle example query and configuration

The only mandatory column in the product table is the (product) ID column. If no additional barcode column is used, this column should be a VARCHAR string and marked as the PRIMARY KEY.

Always make sure to configure the 'Unique ID' field in the 'Data'-tab as well.

It is recommended to name the [ID] column of the link table the same as the [ID] column of the product table.

The names and formats can be changed as desired. An example of a product table is shown below.

```
CREATE TABLE products (
[ID] VARCHAR(20) PRIMARY KEY,
[Description] VARCHAR(60),
[Group] VARCHAR(10),
[Std. Price] VARCHAR(10),
[Sell Price] VARCHAR(10),
[Content] VARCHAR(20),
[Unit] VARCHAR(10))
```

If barcodes, like EAN/UPC, are used for linking, then be aware that **multiple EAN/UPC barcodes can exist for the same product(!)** making them unsuitable as 'Unique ID'.

In this situation the barcode column would be the 'PRIMARY KEY', but the product ID- column must be configured as 'Unique ID'.

An example of a product table with barcodes is shown below.

```
CREATE TABLE products
(
[Barcode] VARCHAR(20) PRIMARY KEY,
[ID] VARCHAR(20,
[Description] VARCHAR(60),
[Group] VARCHAR(10),
[Std. Price] VARCHAR(10),
[Sell Price] VARCHAR(10),
[Content] VARCHAR(20),
[Unit] VARCHAR(10)
)
```

The configuration in the 'Data'-tab would then become:

Barcode	Barcode ▼
Unique ID	ID ▼
Description	Description ▼

4.2.5. Product staging table

Changed product information should be inserted in the product staging table and the ESL server will automatically detect and process this data and insert the data into the Product table. If the processed data requires an ESL to be updated, it will automatically generate a new image and sent it to the ESL.

The product staging table must have the exact same formatting as the product table, but should also contain a 'DELETE' column. The 'DELETE' column makes it possible to remove a product from the link table, by placing an 'X' in this column.

The product staging table that match the previous product table examples are shown below.

```
CREATE TABLE products_staging
(
```

```
[ID] VARCHAR(20) PRIMARY KEY,
[Description] VARCHAR(60),
[Group] VARCHAR(10),
[Std. Price] VARCHAR(10),
[Sell Price] VARCHAR(10),
[Content] VARCHAR(20),
[Unit] VARCHAR(10),
[DELETE] VARCHAR(1))
```

Or with barcode column:

```
CREATE TABLE products_staging
(
[Barcode] VARCHAR(20) PRIMARY KEY,
[ID] VARCHAR(20),
... )
```

4.2.6. ESL status and configuration table

When using SQL, the ESL server automatically creates a table called 'labelstatus'. This table is used to store all information about each ESL in the server and has the following table design.

Column Name	Data Type	Allow Nulls
ID	varchar(40)	☑
⚷ MAC	varchar(16)	☐
[GROUP]	varchar(50)	☑
DESCRIPTION	varchar(100)	☑
IMAGE_FILE	varchar(80)	☑
POLL_INTERVAL	int	☑
POLL_TIMEOUT	int	☑
SCAN_INTERVAL	int	☑
BATTERY_STATUS	int	☑
BATTERY_VOLTAGE	decimal(5, 2)	☑
VARIANT	varchar(5)	☑
FIRMWARE_VERSION	varchar(10)	☑
FIRMWARE_SUBVERSION	varchar(10)	☑
IMAGE_ID	int	☑
IMAGE_ID_LOCAL	int	☑
BACKLIGHT	int	☑
DISPLAY_OPTIONS	int	☑
LQI	int	☐
LQI_RX	int	☑
LAST_POLL	datetime	☑
LAST_INFO	datetime	☑
LAST_IMAGE	datetime	☑
BASE_STATION	varchar(25)	☑
SCAN_CHANNELS	int	☑
STATUS	int	☑
FIRMWARE_STATUS	int	☑
IMAGE_STATUS	int	☑
X	int	☑
Y	int	☑
WIDTH	int	☑
HEIGHT	int	☑
IMG_IX	int	☑
IMG_FORMAT	int	☑
BOOT_COUNT	int	☑
TEMPERATURE	int	☑
LANID	varchar(4)	☑

Below an overview is given of the different fields in the 'labelstatus' table.

ID	Unique article ID to which the ESL is linked. <Null> or empty if the ESL is/was present in the ESL system, but not linked.
MAC	16-digit hexadecimal IEEE (MAC) address of the ESL.
GROUP	If a group field is configured in the Data-tab, this field will contain the group of the product linked to the ESL
DESCRIPTION	If the ESL is linked, this field will contain the configured product description. <Null> if not linked or unknown
IMAGE_FILE	If the ESL is linked, this field will contain the file name of the automatically generated image content. <Null> if not linked.
POLL_INTERVAL	If the ESL is physically present, this field will contain the poll

	interval of the ESL. <Null> if the poll interval is unknown
POLL_TIMEOUT	If the ESL is physically present, this field will contain the poll timeout of the ESL. <Null> if the poll timeout is unknown
SCAN_INTERVAL	If the ESL is physically present, this field will contain the scan/associate interval of the ESL. <Null> if the interval is unknown.
BATTERY_STATUS	If the ESL is physically present, this field will contain the battery status of the ESL. 1=OK, 0=Low, <Null> = Unknown
VARIANT	If the ESL is physically present or linked, this field will contain the software variant of the ESL. 'H'=EE200/EE201 and 'I'=EE201. <Null> = Unknown
FIRMWARE_ VERSION	This field will contain the software version of the ESL. Typically in the format: IBxVyyyy. <Null> = Unknown
FIRMWARE_ SUBVERSION	This field will contain the software subversion of the ESL. Typically not used by e-paper based ESLs. <Null> = Unknown
IMAGE_ID	This field will contain the image ID of the image shown on the ESL. 0=default image; 1=custom Image; <Null> = Unknown. In the future the ESLs can contain multiple images.
BACKLIGHT	If the ESL is physically present, this field will contain the back light setting (ranging from 5-100(%), <Null> = unknown). Only used by TFT based ESLs
DISPLAY_OPTIONS	If the ESL is physically present, this field will contain the image orientation. 0=Normal, 1=Upside-down, <Null> = unknown
LQI	This field will contain the last received Link-Quality-Index, indicating the connection strength with the base station. ○ -1 Unknown / not connected ○ 0-3 Link quality is poor and high packet loss (>40%) ○ 4-7 Link quality is poor and significant packet loss (>20%) ○ 8-10 Link quality is poor, but no significant packet loss (<20%) ○ >10 Link quality is good and no significant packet loss
LQI_RX	This field will contain the last received Link-Quality-Index measured by the ESL, indicating the connection strength with the base station. Ranges from 1-255 is fine, 0 is poor, <Null>=unknown.
LAST_POLL	This field will contain the UTC time of the last received poll for data. <Null> means unknown / never received. *(This field is currently not updated on each poll to reduce the amount of SQL queries executed by the ESL server).*
LAST_INFO	This field will contain the UTC time of the last reported status by the ESL. <Null> means unknown / never received.
LAST_IMAGE	This field will contain the UTC time of the last successful image update. <Null> means unknown or hasn't been updated.
BASE_STATION	If the ESL is/was connected to a base station, this field will contain the IP address of the last base station it was connected to. String format is xxx.xxx.xxx.xxx (i.e. 192.168.0.12). <Null> = unknown or never connected. Use the 'STATUS' field to verify if the ESL is currently connected or not.
SCAN_CHANNELS	This field contains an integer value of which each bit represents an RF-channel on which the ESL will or will not search for base stations (when not connected).

	Example: 101746688 would represent a binary value of '11000010001000100000000000'. Counting from right to left this means, channels 11, 15, 20, 25 and 26 will be searched. \<Null\> = unknown.

STATUS	This field contains an integer value representing the last reported status of the ESL.
	`<NULL> = Unknown`
	`0 = ASSOCIATED`
	`1 = POLLING`
	`2 = INFO RECEIVED`
	`3 = DISASSOCIATED`
	`4 = COMMAND COMPLETED`
	`5 = COMMAND NOT_ACCEPTED / DISGARDED`
	`6 = POLL TIMEOUT`
	`7 = COMMAND_TIMEOUT`
	`8 = ADDRESS CHANGED (factory status only)`
	`9 = DEACTIVATED`
	`10 = IMAGE_UPDATE_COMPLETED`
	`11 = IMAGE_UPDATE_FAILED`
	`12 = FIRMWARE_UPDATE_COMPLETED`
	`13 = FIRMWARE_UPDATE_STARTED`
	`14 = FIRMWARE_UPDATE_FAILED`
	`15 = REACTIVATED`

FIRMWARE_STATUS	This field contains an integer value representing the last reported status of the ESL.
	`<NULL> = Unknown`
	`0 = Idle`
	`1 = Firmware update pending`
	`2 = Firmware update queud`
	`3 = <not used for firmware updates>`
	`4 = Firmware update failed`
	`5 = <not used for firmware updates>`
	`6 = <not used for firmware updates>`
	`7 = <not used for firmware updates>`
	`8 = Firmware update succesfully completed`

IMAGE_STATUS	This field contains an integer value representing the last reported image status of the ESL.
	`<NULL> = Unknown`
	`0 = Idle`
	`1 = Image update pending`
	`2 = Image update queued`
	`3 = Image update queued and next already pending`
	`4 = Image update failed`
	`5 = Image update failed, waiting for retry`
	`6 = Image update failed, busy retrying`
	`7 = Image update failed, busy retrying, next pending`
	`8 = Image updated succesfully`
	`16 = Image error (image could not be opened, found or generated)`

X, Y	If the ESL is linked, these fields contain an integer value of the X-, Y-offset of the generated image on the ESL. Generally these values will be '0'. \<Null\> = Unknown or not linked
WIDTH, HEIGHT	If the ESL is linked, these fields contain an integer value of the width and height (in pixels) of the generated image on the ESL. Generally these values will be 200 x 96 for the EE200/1.

	<Null> = Unknown or not linked
IMG_IX	This field will generally be '0'. Only if multiple products will be shown on a single large ESL, then IMG_IX will be different for each image that is displayed on the ESL. <Null> = Unknown or not linked
IMG_FORMAT	If an e-paper ESL is linked, this field will generally be set to '1' indicating that the image is transmitted compressed to the ESL. If the data is sent uncompressed it would be '0'. Other values are reserved for TFT based ESLs (TIFF=2, JPG=3, GIF=4, FIRMARE=5). <Null> = Unknown or not linked
BOOT_COUNT	Shows how many time the firmware of the ESL was reset
TEMPERATURE	Shows the temperature range on which the ESL is supposed to run on (0=default; higher values are to compensate for low temperatures). Please note that low temperatures and values > 0 will increase power consumption!
LANID	LANID = 0000 means ESL will associate to all base stations. LANID <> 0000 means ESLs will only associate to base stations with the same LANID or with LANID 0000

4.2.7. Base station Status table

When using SQL, the ESL server automatically creates a table called 'basestationstatus'. This table is used to store all information about each EBSxx base station in the server application and has the following table design

	Column Name	Data Type	Allow Nulls
🔑	MAC	varchar(17)	☐
	IP_ADDRESS	varchar(15)	☑
	PORT	int	☑
	MODEL	varchar(10)	☑
	NAME	varchar(25)	☑
	PAN_ID	varchar(4)	☑
	CHANNEL	int	☑
	VERSION	varchar(10)	☑
	STATUS	varchar(15)	☑
	ESLS	int	☑
	MIN_LOAD	int	☑
	MAX_LOAD	int	☑
	LAN_ID	varchar(4)	☑

MAC	12-digit hexadecimal MAC address of the base station
IP_ADDRESS	Current IP address of the base station
PORT	Current IP port number of the base station
MODEL	Current hardware model of the base station (EBS30 or EBS31)
NAME	Current local name f the base station
PAN_ID	Current PAN-Id on which this base station operates
CHANNEL	Current RF-channel on which this base station operates
VERSION	Firmware version of the base station
STATUS	Current connection status (Connected or Disconnected)
ESLS	Current number of associated ESLs to this base station
MIN_LOAD	RFU (for load balancing)
MAX_LOAD	RFU (for load balancing)
LAN_ID	LANID = 0000 (default) means all ESL will associate with this base station. LANID <> 0000 means only ESLs with the same LANID or with LANID=0 will associate with this base stations

4.2.8. Logging table

When using SQL, the ESL server automatically creates a table called 'esllog. This table is used to store all alerts and events, except user activity. This table has the following table design

Column Name	Data Type	Allow Nulls
🔑 ID	int	☐
BASESTATION	varchar(25)	☑
ESL	varchar(16)	☑
TIME	datetime	☑
MESSAGE	varchar(256)	☑

4.2.9. Change log table

When using SQL, the ESL server automatically creates a table called 'changelog. This table is used to store user activity. This table has the following table design

Column Name	Data Type	Allow Nulls
🔑 ID	int	☐
[USER]	varchar(80)	☑
TIME	datetime	☑
MESSAGE	varchar(255)	☑
[FROM]	varchar(255)	☑
[TO]	varchar(255)	☑
BASESTATION	varchar(25)	☑
ESL	varchar(16)	☑

This table is used for by the ESL server and the Web application.

4.2.10. User table

When using SQL, the ESL server automatically creates a table called 'user. This table is used to store usernames, encrypted password and access rights.

Column Name	Data Type	Allow Nulls
🔑 [USER]	varchar(64)	☐
PASSWORD	varchar(64)	☑
RIGHTS	varchar(16)	☑

This table is used for user access control by the ESL server and the Web application. Users can only be added, removed or altered in the ESL server application.

4.2.11. (deprecated) Product look-up table

This table is only required when the 'Web-application for linking' is used and no barcode column

has been added to the 'Product' tables.

A product look-up table can be added for easy linking of products to ESLs by scanning the barcode of a product followed by the address barcode on the ESL. This table should only contain the bar codes and related unique product IDs and a short description of the product.

A typical product look-up table would be created using the following query. The use of the barcode column as primary key makes it possible to have products with multiple barcodes.

```
CREATE TABLE barcodes
(
[Barcode] VARCHAR(40) NOT NULL PRIMARY KEY,
[ProductID] VARCHAR(40) NOT NULL,
[Description] VARCHAR(50),
)
```

Important notes:
Please make sure that the barcodes in this table are in the same format as output format of the handheld barcode reader. Databases often strip any leading zeros and/or checksums, while barcode readers DO output them. This can cause barcodes of products not to be recognized.

When using the supplied WebApp or using a barcode terminal to link ESLs, ALL barcodes in the look-up table should NOT have any leading zeros.

4.2.12. ESL Server SQL configuration

After creating the required SQL tables it is possible configure the ESL server application. Go to the 'Data'-tab to start the configuration.

The first step is selecting which type of SQL database. This can be MySQL, MS-SQL and Oracle.

Next fill in the SQL Server address, login, password and database name and press 'Test' to verify the connection with the SQL server is configured correctly.

After a successful connection to the SQL server select the two product tables and two link tables.

The other tables will be generated automatically, but can be renamed if desired.

After that, select which fields of the product table are the 'Barcode', 'Unique ID', 'Group' and 'Description' field. (The sell and standard price are used only by the web application)

Finally the currency and culture info can be configured in the 'settings' tab

Configuration of the Web application is described in Chapter 8.

Press 'Apply' again to finish the configuration.

After a successful configuration, it is now possible to start adding products to the product staging table using standard SQL queries.

An example of a query to add a product to the product staging table would look like this:

```sql
CREATE TABLE products_staging
(
[ID] VARCHAR(20) PRIMARY KEY,
[Description] VARCHAR(60),
[Group] VARCHAR(10),
[Std. Price] VARCHAR(10),
[Sell Price] VARCHAR(10),
[Content] VARCHAR(20),
[Unit] VARCHAR(10),
[DELETE] VARCHAR(1)
)
```

```
INSERT INTO [products_staging] (
[ID],
[Description],
[Group],
[Std. Price],
[Sell Price],
[Content],
[Unit],
[DELETE] )
VALUES
( '031',
'HOUSE WINE RED',
'30',
'3,00',
'3,00',
'2',
'LT',
''
)
```

To delete a product, use 'X' as value for the [DELETE] field.

An example of a query to add a new link to the link staging table would look like this:

```
INSERT INTO Links_Staging ([ID], [Variant], [MAC], [DELETE])
VALUES ('008', 'H', '00124B00010DA031', '');
```

To delete a link, use 'X' as value for the [DELETE] field and [ID] and [Variant] can be left empty.

```
INSERT INTO Links_Staging ([ID], [Variant], [MAC], [DELETE])
VALUES (NULL, NULL, '00124B00010DA031', 'X');
```

Links can also be added manually, which is described in the following chapters.

It's also possible to create links by using the supplied web app for handheld terminals like the H22, which allows quick linking by scanning the barcode on a products and the barcode on the ESL. This requires a web-server running the web-app and a barcode field to be present in the product table or otherwise a separate product look-up table to be present. See chapter 8 for more information.

4.3. CSV / SQL Hybrid configuration

When using a CSV / SQL hybrid solution the product databases and ESL-product links are provided by CSV-files, but all ESL and Base station status information are logged into an SQL database. This makes it easier to link the back-office to the ESL server using simple CSV-files and all status information is logged into an SQL database for external monitoring and trouble shooting. Also the start-up time of the ESL server will be significantly faster when a large number of ESLs are linked compared to using CSV only.

To make such a configuration, start by configuring the ESL server with a standard CSV configuration as described in chapter 4.1.

When done, go to the 'Settings'-tab and change the 'database format' for logging to 'MS- SQL', 'MySQL' or 'Oracle' as shown below.

After that, go back to the 'Data'-tab and temporarily change the 'Database format' to 'MS- SQL', 'MySQL' or 'Oracle' as shown below.

Configure all SQL options as described in the previous chapter, but clear the product and link table fields.

When done, the SQL table configuration should look like this:

To test the SQL configuration, press the 'Test' button. If the connection is OK, then press 'Apply' to save the SQL configuration.

The various SQL tables will be automatically created.

After that, change back the 'Database format' to 'CSV' and press 'Apply' again to finish the configuration.

OPTICON

4.4.   External content management configuration

When using the external content management solution the 3rd party content management system handles the generation of the images and the linking of products to ESLs.

The ESL server application monitors an input folder for new images and distributes them to the corresponding base stations / ESLs.

The only main database configurations for this set-up in the ESL server are the 'Input' and 'Output' folder (select CSV-database format). The other configurations will be ignored.

If the 'Output' folder already contains the files 'dbase.csv' or 'linked.csv' then these files should be removed, so the system won't use them as product or link database

Optionally, status information can be logged into a SQL database using the same configuration as described in chapter 4.3 CSV / SQL Hybrid configuration.

If desired, remote control over the base stations and ESLs can be achieved using SQL queries as described in Appendix E: Using the SQL API

When the images for ESLs are generated and supplied by the content management system, then they must meet the following criteria:

- The image format must be either .bmp, .gif or .png

- The images must be monochrome (no gray scale or anti-aliasing allowed!).

- Each image file must be uniquely named for each ESL (it's not allowed to use the same image file for multiple ESLs)

- It's recommended to include the MAC-address of the ESL in the image filename (i.e. 001_3889DC0000002345.bmp)

In order for image files to be automatically processed by the ESL server they should be placed in the configured 'Input'-folder, accompanied by a matching xml-info file.

The xml-info file provides the ESL server application with information on which image should be shown on which ESL and some other optional information which is displayed in the ESL server application.

The file name of the xml-info file is predetermined to:

```
<Unique product ID>_<MAC address>.XML (i.e. 001_3889DC0000002345.xml)
```

You can find such xml-files in the 'Output'-folder of your installation, which can be used as reference on how to generate them.

Example of an xml-info file:

```xml
<?xml version="1.0" encoding="utf-8"?>
<EslImageInfo xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <ImageFile>001_3889DC0000003441.bmp</ImageFile>
  <X>0</X>
  <Y>0</Y>
  <HOffset>0</HOffset>
  <VOffset>0</VOffset>
  <Variant>H</Variant>
  <Group>Vegetables</Group>
  <Note>YELLOW CARROTS</Note>
  <Label>3889DC0000003441</Label>
  <Width>200</Width>
  <Height>96</Height>
  <Ix>1</Ix>
  <Format>1</Format>
</EslImageInfo>
```

The variable fields are marked in bold and the functions of these fields are described on the next page.

<ImageFile>	:	Must match the image file name for this ESL (.bmp, .gif or .png)
<Variant>	:	EE200 / EE201 firmware variant: H or I
<Group>	:	Group field is intended to supply additional information to the user on the product group in the 'ESL'-tab (useful for sorting or searching)
<Note>	:	Note field is intended to supply additional information to the user about the product in the 'ESL'-tab (useful for sorting or searching)
<Label>	:	16-digit MAC of the ESL for which the image is intended
<Width> / <Height>	:	EE200 / EE201 image size is 200 x 96 pixels
<Ix>	:	Used by the ESL server to keep track which image is currently displayed on the ESL. It should increment from 1 to 255, 1 to 255, etc. 0=reserved for the default image. Not incrementing the image index can lead to incorrect images on the ESL display if a new image update for the same ESL starts before the previous update was finished.
<Format>	:	Determines the image transfer protocol, for the EE200/EE201 it should be fixed to '1'. (It's not related to using bmp, png or gif)

4.4.1. Unlinking

To unlink an ESL, place an XML-file into the 'Input'-folder with following name format:

<MAC address>.XML

To indicate this ESL should be **unlinked** leave the ImageFile-field empty:

<ImageFile></ImageFile>

4.5. Manual editing of the product database

Using the 'Data'-tab of the ESL server application it is possible to quickly view and edit the content of the current product database. To do this, select the field that you wish to change and edit the value.

After editing, press the 'Save Changes' button in the lower-left corner to apply the changes. The changes will be automatically processed and send to the ESLs if the resulting image is altered. To undo any unsaved changes, press the 'Reload' button to reload to the original database.
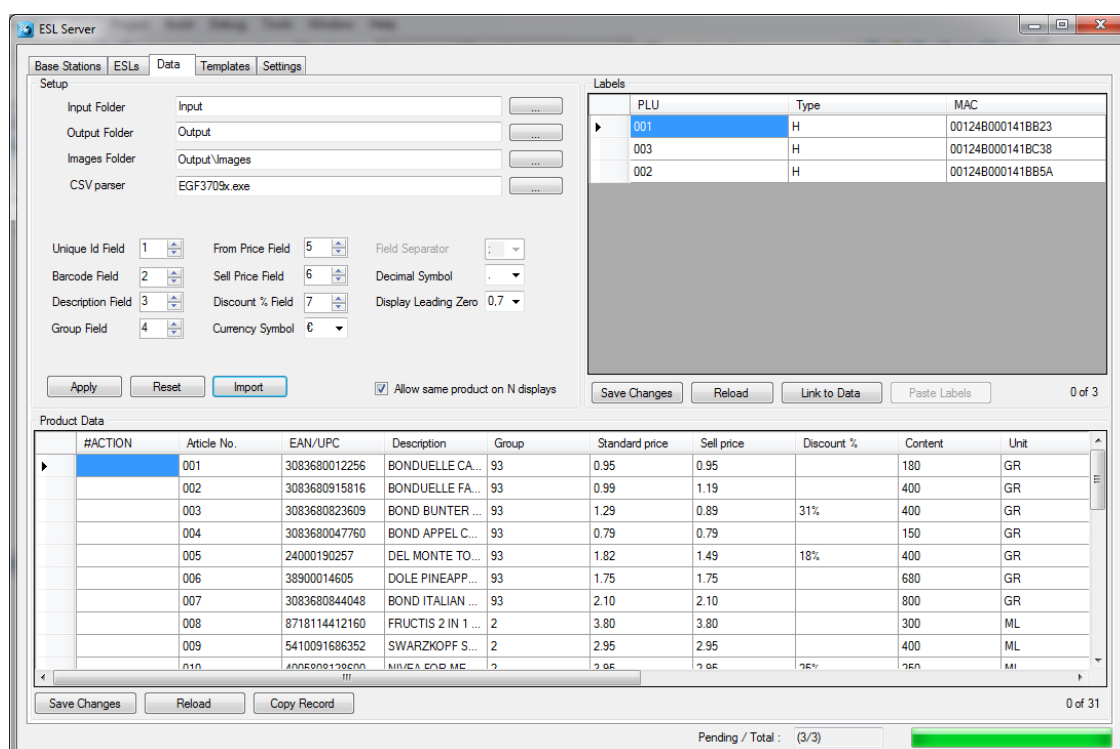
4.6. Manual editing of the product-ESL link database

In the top-right corner of the 'Data'-tab, the database containing the links between product and ESLs can be reviewed and edited.

To edit existing links, simply change the fields by selecting the field and changing its content.

To add new product-ESL links, first select the ESLs that you wish to link in the 'ESLs'-tab and then press the 'Copy Labels' button. After that, go back to the 'Data'-tab and press the 'Paste Labels' button. This will cause the copied labels to be pasted in the link database.

To quickly link products to the added ESLs, select both the row(s) of the ESL(s) as well as the row(s) of the product(s) and press the 'Link to Data'-button.

To apply the changes press the 'Save Changes' button. The changes will be automatically processed and send to the ESLs. To undo any unsaved changes, press the 'Reload' button to reload to the original link database.

5. Designing the ESL image layout

Probably the most important step in the customization process of the ESL system is determining the desired end-result, being the images that should be displayed on the ESLs.

When designing the layout of an e-paper ESL, it's important to know the limitations of what can and what can't be displayed.

The main characteristics are:

- Fixed resolution of 200x96 pixels
- Monochrome images only (black/white)
- No anti-aliasing can be used on fonts (limiting the amount of available fonts)
- Monochrome company logos and other product (group) images can be added to the layout.
- 1D- and 2D-Barcodes can be added using information from the database
- Texts, images and barcodes can be rotated only in 90° angles
- Dynamic texts, using information from the database, can be added and aligned as desired.
- Different templates for groups of ELSs is possible by adding objects to the template using   a template selection file with conditions (i.e. condition: 'field x' (group) equals '5')

The layout of an ESL images is specified using template files, which are XML-based. The default location of the templates is "./Templates".

The ESL template files can be edited using the **graphical template editor** and using an **XML-editor**. Both options are available in the 'Templates' tab to design your own image layout.

The ESL server application uses 3 types of templates.

- A **Template Selection File** specifies which image template is used under which conditions. Using a template selection file makes it possible to apply different templates to different models of ESL (i.e. 2" and 2.7") and to different product groups or conditions (i.e. discounts).

- **Image templates** specify the actual image layout of the images that will be generated.

- **Images Folder**. All automatically generated images and XML-files will be placed in this folder. Once these images and XML-files have been sent to ESLs they will be moved to the sub-folder: '/Completed'

- An **information template** is used to generate the XML-files that provide the ESL server application with all the necessary information about the linked ESL and product (Unique ID, MAC address and description). By default this template file is automatically generated using the database configuration in the 'Data'-tab. Only if you wish to add multiple articles on the same ESL, then adding this template will be necessary. Please contact Opticon technical support for a detailed description on how to do this.

5.1. Editing the Template Selection File

As mentioned above, the template selection file specifies which image template is used under which conditions. If a template selection file is used, then it will be configured as 'Default template' in the 'Templates'-tab.

If only one image template is used, then it's not necessary to use a template selection file. So the 'Default template' can simply be set to this image template file.

To edit the template selection file, it first needs to be opened by selecting it under 'Found templates'. After that the file will be displayed on the right.

```xml
1    <?xml version="1.0" encoding="utf-8"?>
2    <EslLayoutInfo>
3        <EslTemplate>
4            <Variant>HI</Variant>              <!-- 'H' = EE200 / 'I' = EE201 -->
5            <Condition>%7%=</Condition>        <!-- No discount -->
6            <FileName>Normal_2.xml</FileName>
7        </EslTemplate>
8        <EslTemplate>
9            <Variant>HI</Variant>              <!-- 'H' = EE200 / 'I' = EE201 -->
10           <Condition>%7%</Condition>         <!-- With discount -->
11           <FileName>Discount_2.xml</FileName>
12       </EslTemplate>
13       <EslTemplate>
14           <Variant>HI</Variant>              <!-- 'H' = EE200 / 'I' = EE201 -->
15           <Condition></Condition>            <!-- Article not found -->
16           <FileName>NotFound_2.xml</FileName>
17       </EslTemplate>
18       <EslTemplate>
19           <Variant>K</Variant>               <!-- 'K' = EE270 (2.7" e-paper) -->
20           <Condition>%7%=</Condition>        <!-- No discount -->
21           <FileName>Normal_2_7.xml</FileName>
22       </EslTemplate>
23       <EslTemplate>
24           <Variant>K</Variant>               <!-- 'K' = EE270 (2.7" e-paper) -->
25           <Condition>%7%</Condition>         <!-- With discount -->
26           <FileName>Discount_2_7.xml</FileName>
27       </EslTemplate>
28       <EslTemplate>
29           <Variant>K</Variant>               <!-- 'K' = EE270 (2.7" e-paper) -->
30           <Condition></Condition>            <!-- Article not found -->
31           <FileName>NotFound_2_7.xml</FileName>
32       </EslTemplate>
33       <EslTemplate>
34           <Variant>Q</Variant>               <!-- 'Q' = EE440 (4.4" e-paper) -->
35           <Condition>%7%=</Condition>        <!-- No discount -->
```

Each listed image template in the selection file will have 3 fields:

- **Variant** Specifies the hardware variant character of the ESL. This character is the 3rd character of the Firmware version of an ESL, which is displayed in the 'ESLs'-tab.

  The current list of supported hardware variants:

  ```
  H = EE200 / EE201 (=2" E-papers)*
  K = EE270 / EE271 (=2.7" E-paper)
  L = ET1000 (=10" TFT Screen)
  F = EB4000 (=40" Bistable LCD e-rail)
  P = ET3200 (=32" TFT Screen)
  ```

Multiple variants can be added to a single template if the image dimensions are the same, i.e. the EE200 and EE201.

- **Condition (optional)** Specifies under which condition the template is used.

    Examples:

    ```
    <!—- Template is used if field '3' contains a value -->
    <Condition>%3%</Condition>

    <!-- Template is used if field '3' equals '12' -->
    <Condition>%3%=12</Condition>

    <!-- Template is used if field '3' equals field '4' -->
    <Condition>%3%=%4%</Condition>

    <!-- Template is used if field '3' is empty -->
    <Condition>%3%=</Condition>

    <!-- Template is used if product was not found -->
    <Condition></Condition>

    <!-- Template is used if field '3' < field '4' -->
    <Condition>%3%<%4%</Condition>

    <!-- Template is used if field '3' >= field '4' -->
    <Condition>%3%>=%4%</Condition>
    ```

- **File name**   File name of the image template (use relative paths to the configured 'Template folder')

After editing the template selection file, press the 'Save' button to make sure that the template file is valid. If not, follow the instructions underneath the XML-editor and try again.

To apply the changes to the ESLs, press the 'Export' button.

5.2.   Editing the Image layout templates

Image layout templates are XML-based and can be edited to design your own label layout.

To edit an image layout template, open it by selecting it under 'Found templates'. After that the template can be edited graphically and by editing the XML-content of the file on the right.

A preview of the current template is shown on the left, which will be updated each time that you press the 'Save' button (or press Ctrl-S).

If the modified template contains an error, then no preview will be displayed and an error message will be shown underneath the editor with detailed information about what's wrong.

To apply the changed templates to the ESLs, press the 'Export' button.

5.2.1. Using Example data

Using the 'Example data' text boxes in between the image preview and the editor it is the possibility to preview the image template with actual product data.

If you leave these 'Example data' fields empty, then the previewer of the Image template will show its text 'as is' in the previewer.

You can either change the 'Example data' manually or you can select one row in the 'Data'-tab and press the 'Copy Record' button to use the selected product as example data. This way you can be sure the generated image will be correct.

Image preview with example data

5.2.2. Using the graphical Template editor

The graphical template editor is selected by pressing on the 'Image'-tab on the right.

Once an existing template is selected under 'Found templates' or a new template is created (Select '<Add>'), the template will be shown on the right.

Image templates currently support 7 types of objects that can be added to design the ESL layout:

A	Text (fixed and dynamic)
/	Lines
▢	Rectangles (filled / non-filled)
◯	Ellipses (filled / non-filled)
‖‖‖	Barcodes
✿	Images (monochrome; fixed and dynamic)
🕐	Time stamp (of the last image update)

To add a new object to the template, select the type of object you want to add, by pressing the corresponding button.

Once the type object is selected, the object can be added by drawing a rectangle on the image below to indicate the position of the new object. After that its properties can be changed as desired..

To edit an existing object left-click on the border or center of the object.

Objects can also be moved and resized by dragging their borders, corners and centers.
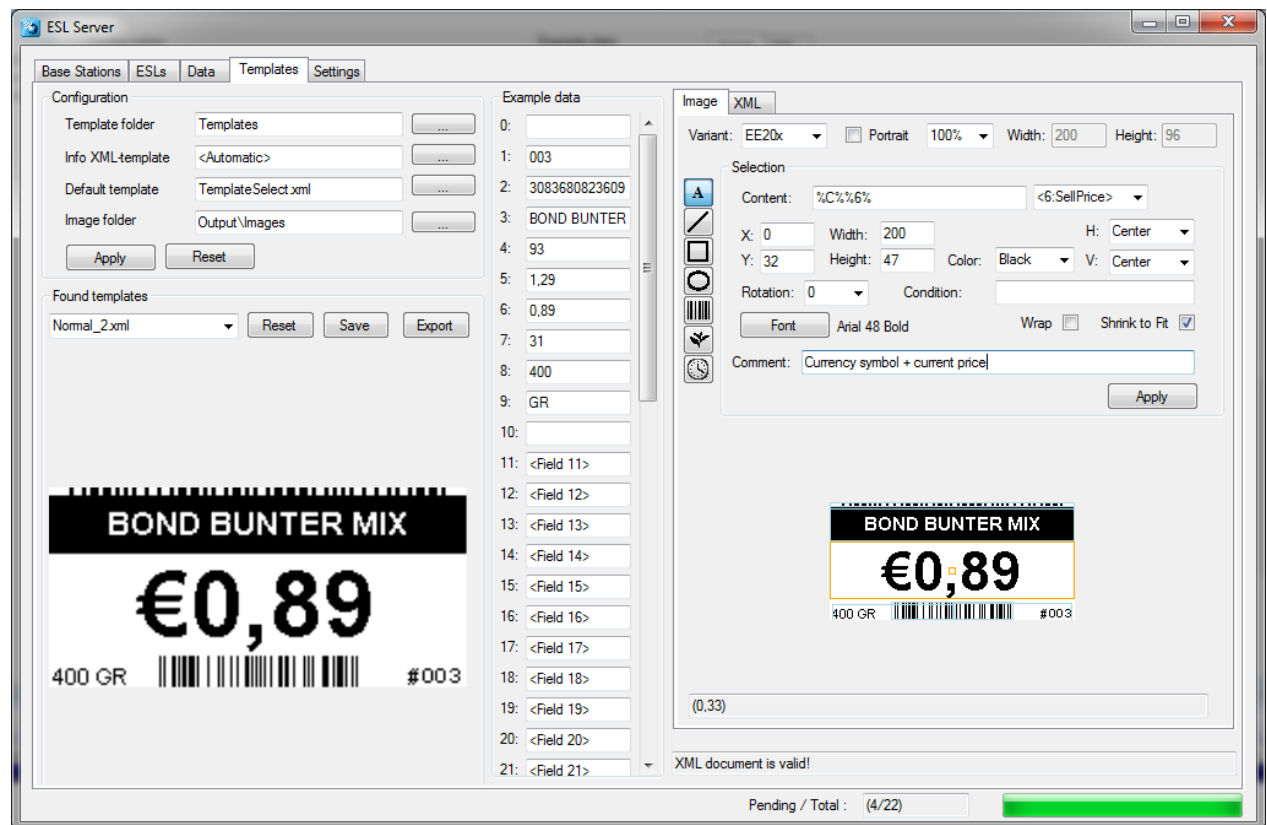
5.2.3. Static and dynamic Content

Static content of barcodes, images and text can be added using the 'Content' textbox.

For example:

Dynamic content of texts, barcodes and images can also be inserted to your template.

To add a certain field of a database record to your template, add the field number between two '%'-characters in the content field of the object.

For ease of use, the field selector can be used to select the desired field number
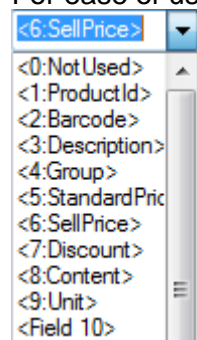
For example, if the CSV-file or SQL-database has the following field sequence:

```
<Modification code>;<Barcode>;<Unique ID>;<Price>;<Description>;<Weight>;<Unit>;
```

Then the price field would be added by using %3%, since it is field 3 (field 0 is the first field).

Content:	%3%

It's also allowed to combine multiple fields as dynamic content.

Example (to add the weight plus standard unit):

Content:	Weight: %5% %6%

You can also use the selected **Currency Symbol** (as specified in the 'Data'-tab) by simply using **%C%** as a shortcut to add the currency symbol.

Content:	%C%%6%

A barcode containing the (short) MAC address of the ESL can be added to your template using %MAC% as content and selecting symbology 'CODE39'.

Content:	%MAC%

(Note: The full MAC address can be added as 'Text' using '%M%')

5.2.4. Conditional objects

Any type of object can be added to the template conditionally, meaning that it is only shown in the resulting image if a certain field in a database record contains any value or equals a certain value.

This is can be done using the 'Condition' property.

Examples:

```
Object is only displayed if field '3' contains any value (not empty):
```

Condition:	%3%

```
Object is only displayed if field '3' equal '12':
```

Condition:	%3%=12

```
Object is only displayed if field '3' equals field '4':
```

Condition:	%3%=%4%

Object is only displayed if field '3' is empty:

Condition: %3%=

(Note: alternatively use <, <=, >= or > for 'larger' and 'smaller than' conditions, but note that when editing XML, it uses '&lt' for a '<' and '&gt' for a '>' symbol)

5.2.5. Using the XML-template designer

The XML-editor is selected by pressing on the 'XML-tab on the right.

The XML-editor shows the XML-code that is being generated from the graphical template editor and consists of a list of objects that have been added to the template using the following formatting:

```
<?xml version="1.0" encoding="utf-8"?>
<EslLayoutInfo>
  <EslTemplate>
        <Variant>H</Variant>              <!-- 'H' = EE20x -->
        <Width>200</Width>
        <Height>96</Height>
        <Object>
                <Text>
                       …
                </Text>
        </Object>
        <Object>
                <Rectangle>
                       …
                </Rectangle>
        </Object>
        <Object>
                <Image>
                       …
                </Image>
        </Object>
        <Object>
                <Line>
                       …
                </Line>
        </Object>
        <Object>
                <Barcode>
                       …
                </Barcode>
        </Object>
  </EslTemplate>
</EslLayoutInfo>
```

The image will be generated by parsing the template from top to bottom, so if one object needs to be drawn on top of another, then this can be done by changing the sequence of the objects in the image template.

The **'Variant'** field specifies the ESL hardware variants to which the image template applies.

The current list of supported hardware variants:

```
H/I/D = EE20x (2" E-papers)*
P = ET3200 (=32" TFT Screen)
T = EE150 (preliminary 1.5" e-paper)
V = EE290 (preliminary 2.9" e-paper)
Q = EE4xx (4.4" e-paper)
X = EE580 (preliminary 5.8" e-paper)
R = EE740 (7.4" e-paper)
```

* Multiple Variants can be added to a single template if the image dimensions are the same, i.e. the EE200 and EE201 both have 200x96 images.

The dynamic content of texts, barcodes and images can be inserted to your template from the

database. To add a certain field from a database record to your image, add the field number between two '%'-characters in the content field of the object.

For example, if the CSV-file has the following field sequence:

```
<Modification code>;<Barcode>;<Unique ID>;<Price>;<Description>;<Weight>;<Unit>;etc. etc.
```

Then the price field would be added by using %3%, since it is field 3 (field 0 is the first field).
It's also allowed to combine multiple fields as dynamic content.

Example (to add the weight plus standard unit):

```
<Content>Weight: %5% %6%</Content>
```

Special characters can be added using the standard XML notation, so for instance an Euro sign can be added using: `€` (since its Unicode is U+20AC).

Alternatively, you can also use the selected **Currency Symbol** (as specified in the 'Data'-tab) by simply using **%C%** as a shortcut to add the currency symbol.

Any type of object can be added to the template conditionally if needed, meaning that it is only shown in the resulting image if a certain field in a CSV-record contains a value or equals a certain value.

This is can be done by adding one of the following lines to any type of object.

```
<!— Object if only displayed if field '3' contains any value -->
<Condition>%3%</Condition>

<!-- Object if only displayed if field '3' equals '12' -->
<Condition>%3%=12</Condition>

<!-- Object if only displayed if field '3' equals field '4' -->
<Condition>%3%=%4%</Condition>

<!-- Object if only displayed if field '3' is empty -->
<Condition>%3%=</Condition>
```

(Note: alternatively use <, <=, >= or > for 'larger' and 'smaller than' conditions, but note that when editing XML, it uses '&lt' for a '<'-symbol and '&gt' for a '>'-symbol)

The following chapters will describe the 6 object types in more detail.

5.2.6. Text

Text can added to the template using the following format:

```
<Object>
 <Condition>%6%</Condition>
 <Text>
      <Content>From: %6%</Content>
      <Color>0</Color>
      <X>0</X>
      <Y>33</Y>
```

```
        <Width>40</Width>
        <Height>20</Height>
        <Font>
                <Type>Arial</Type>
                <Size>9</Size>
                <Italic>0</Italic>
                <Bold>0</Bold>
        </Font>
        <Lines>1</Lines>
        <HAlignment>LEFT</HAlignment>
        <VAlignment>BOTTOM</VAlignment>
        <Rotation>0</Rotation>
        <ShrinkToFit>0</ShrinkToFit>
    </Text>
</Object>
```

- <Condition>   (optional) Displaying conditions
- <Content>   The Dynamic/fixed text to be rendered
- <Color>   Text color (0=Black; 1=White)
- <X> <Y> <Width> <Height>   Specify the rectangle within the text is rendered
- <Font>   Specifies the type, style and size of the used font
- <Lines>   1=No wrapping ; 0 = Wrapping
- <HAlignment>   Horizontal alignment (LEFT, CENTER or RIGHT)
- <VAlignment>   Vertical alignment (TOP, CENTER, BOTTOM)
- <Rotation>   Rotation angle (0, 90, 180 or 270)
- <ShringToFit>   1=Font size is lowered to fit the content; 0=disabled

5.2.7. Rectangles

A rectangle can be added to the template using the following format:

```
<Object>
        <Condition>%6%</Condition>
        <Rectangle>
                <Color>0</Color>
                <X>0</X>
                <Y>0</Y>
                <Width>200</Width>
                <Height>32</Height>
                <Size>0</Size>
        </Rectangle>
</Object>
```

- <Condition>   (optional) Displaying conditions
- <Color>   Color (0=Black; 1=White)
- <X> <Y> <Width> <Height> Specify the position and size of the rectangle
- <Size>   Border thickness (0=filled)

5.2.8. Ellipses

An ellipse can be added to the template using the following format:

```
<Object>
      <Condition>%6%</Condition>
      <Ellipse>
            <Color>0</Color>
            <X>0</X>
            <Y>0</Y>
            <Width>200</Width>
            <Height>32</Height>
            <Size>0</Size>
      </Ellipse>
</Object>
```

- <Condition>   (optional) Displaying conditions
- <Color>   Color (0=Black; 1=White)
- <X> <Y> <Width> <Height>   Specify the position and size of the ellipse
- <Size>   Border thickness (0=filled)

5.2.9. Lines

A line can be added to the template using the following format:

```
<Object>
      <Line>
            <X>16</X>
            <Y>41</Y>
            <Width>64</Width>
            <Height>20</Height>
            <Size>0</Size>
            <Color>0</Color>
            <Rotation>0</Rotation>
      </Line>
</Object>
```

- <Condition>   (optional) Displaying conditions
- <Color>   Color (0=Black; 1=White)
- <X> <Y> <Width> <Height>   Specify the borders of the line
- <Size>   Thickness of the line
- <Rotation>   Rotation angle (0, 90, 180 or 270)

5.2.10. Images

An image (fixed or dynamic) can be added to the template using the following format:

```
<Object>
        <Condition>%4%</Condition>
        <Image>
                <File>Images\G%4%.bmp</File>
                <X>173</X>
                <Y>33</Y>
                <Width>32</Width>
                <Height>32</Height>
                <HAlignment>CENTER</HAlignment>
                <VAlignment>BOTTOM</VAlignment>
                <Rotation>0</Rotation>
        </Image>
</Object>
```

- <Condition>   (optional) Displaying conditions
- <File>   Image file path (absolute or relative to 'Template'-folder)
- <X> <Y>   Specify the start-position of the image
- <Width> <Height>   (optional) Specify the maximum size of the image
- <HAlignment>   (optional) Horizontal alignment (LEFT, CENTER or RIGHT)
- <VAlignment>   (optional) Vertical alignment (TOP, BOTTOM)
- <Rotation>   Rotation angle (0, 90, 180 or 270)

5.2.11. Barcodes

Almost any type of barcode can be added to the template using the following format:

```
<Object>
        <Condition>%4%</Condition>
        <Barcode>
                <Content>%4%</Content>
                <Color>0</Color>
                <Type>5</Type>
                <X>50</X>
                <Y>80</Y>
                <Width>100</Width>
                <Height>20</Height>
                <HAlignment>CENTER</HAlignment>
                <VAlignment>BOTTOM</VAlignment>
                <Rotation>0</Rotation>
        </Barcode>
</Object>
```

- <Condition>   (optional) Displaying conditions
- <Content>   Content of the barcode
- <Color>   Foreground Color (0=Black; 1=White)
- <Type>   Specifies the bar code type (See list below)
- <X> <Y> <Width> <Height>   Specify the rectangle within the barcode is rendered
- <HAlignment>   Horizontal alignment (LEFT, CENTER or RIGHT)
- <VAlignment>   Vertical alignment (TOP, BOTTOM)

The barcode type can be one of the following values:

0. EAN/UPC (auto detect; requires a valid check digit)
1. UPC-A
2. UPC-E
3. UPC 2 Addon/supplemental chars
4. UPC 5 Addon/supplemental chars
5. EAN-13
6. EAN-8
7. Interleaved 2of5
8. Standard 2of5
9. Industrial 2of5
10. Code-39
11. Code-39 Full ASCI
12. Codabar
13. PostNet
14. BOOKLAND
15. ISBN
16. JAN13
17. MSI (Mod10)
18. MSI (Mod10/Mod10)
19. MSI (Mod11)
20. MSI (Mod11/Mod10)
21. Modified Plessey
22. Code-11
23. USD8
24. UCC-12
25. UCC-13
26. LOGMARS6
27. Code-128
28. Code-128 (Code-set A)
29. Code-128 (Code-set B)
30. Code-128 (Code-set C)
31. ITF14 (Interleaved 2of5 14 digits)
32. Code-93
33. Telepen
34. `Aztec`
35. Data Matrix
36. PDF417
37. QR-Code

Please note that the content of a barcode must match various criteria (i.e. fixed length and check digits) in order it to render properly. If a barcode is not (correctly) rendered, then please check the log-file in the sub-folder 'Log' of the configured Output-folder for more information about the cause of the problem (search for 'Barcode rendering failed').

6. Connecting the back office using CSV-files

The back office triggers the ESL server application by placing a file, containing either only product modifications or a complete product database, in the 'Input' folder (see chapter 4).

Only one file may reside in this folder to enforce sequential handling. This means that the Back office may only store another file, when the current file has been processed and moved to a different folder.

To avoid read/write conflicts the database files should first be copied into the specified folder under a temporary name (i.e. *.tmp) and renamed to the correct name afterwards.

The placing and processing of incoming CSV-files should be done as follows:

- The file is moved/copied to a configurable local folder, called the 'Input folder'.
- When moving/copying the file from the Back office to the folder a temporary name is used, i.e. 'productdata_1.tmp', 'productdata_2.tmp', etc.
- After the complete file has been moved/copied, the file will be renamed to its final name, i.e. 'productdata_1.csv', 'productdata_2.csv', etc.
- The server application will pick up the CSV-file as soon as it detects this rename file event.
- The file will be processed instantly, as the records contain no start/end date.
- Once a file is processed, it will be moved to the configurable 'Processed' folder and the Back   office is able to put another file in the folder.
- When errors occur, they will be reported in an error log in the 'Log' folder. The data file will not   be moved in case of errors.
- The ESL server application will call the middleware applications to parse the incoming CSV-file   into a combined database file. After that the ESL server application will generates the information   and image files that are used to update the ESLs.

6.1. Product database format

Currently the only supported database format is CSV (comma separated values) with the following properties:

- UTF-8, UTF-7, ANSI or Unicode encoding (other encoding formats on request)
- Allowed field separator(s): **1) semi-colons**
- Methods: **1) modification database** (add, replace, delete) and **2) complete database**
- Each product in the database must contain a unique product ID field to allow linking of an ESL to a product.
- If there's a 1-to-many relation between the unique product ID and the barcode on this product, then the same product should be added to the database multiple times for each unique barcode, but all other fields must be identical. Also any product modifications (like price) should be applied on each record for this product.
- The adding of a product barcode field to the database isn't mandatory. However it is recommended, since it adds the possibility to quickly link ESLs to a product by simply scanning the barcodes on the product and the ESL using the barcode terminal.
- If any, a column header line should start with '#'.

When using a **modification database**, the modification field should be located at the beginning of each line. **As modification code the characters 'I' (Insert), 'R' (replace) and 'D' (delete) should be used**.

The typical record format of a product modification database would be:

```
<Modification code>;<Unique ID>;<Barcode>;<Description>;<Field 4>;<Field 5>;etc.
```

When a **complete database** is used (meaning any new database file replaces the previous one), the **first (modification) field of each line must be left empty**.

The typical record format of a complete product database would be:

```
;<Unique ID>;<Barcode>;<Description>;<Field 4>;<Field 5>;etc.
```

Any field that should be displayed on the ESL should be in the same formatting as it should be displayed on the ESL, because the image generator uses the exact field data to render the images.

Note: Database formats other than CSV can be supported, but require modifications to the existing middleware application. Please contact Opticon for more information on supporting different database formats.

6.2. Product-ESL link database format

The link between an ESL label and a product is made by linking the unique MAC address of the ESL label to the unique product ID.

To create, modify or delete the product-ESL links various methods can be used:

• Using a barcode terminal can be used (see chapter 7)

• Using the 'Data'-tab of the ESL server application (see chapter 4.6).

• Using a custom method (i.e. using an existing cash-register or terminal).

All of the above methods work by placing the product-ESL link database in the configured 'Input' folder, which is then automatically processed by the middleware application.

This link database is a CSV-file that must contain the word **'link'** in its filename and has the extension **'.csv'** (i.e. 'Product_links1.csv').

Each record of a **complete link-file** (to replace the existing link database) should have the following exact format:

```
<Unique ID>;<Hardware variant>;<16-digit MAC address>
```

The current list of supported hardware variants is:

```
H,I,D = EE200 / EE201 (=2" E-paper)
K = EE270 / EE271 (=2.7" E-paper)
L = ET1000 (=10" TFT Screen)
```

Example:

```
012;H;00124B00010DACCD
014;H;00124B00010DA152
008;H;00124B00010DD7C5
013;H;00124B00010DA234
011;K;00124B00010DAF28
010;K;00124B00010DB222
009;K;00124B00010DC9E1
015;H;00124B000141BB01
016;H;00124B00010DCF40
017;H;00124B00010DAF3E
018;H;00124B000141B84B
```

Each record of the **modification link-file** should have the following exact format:

```
<Modification code>;<Unique ID>;H;<16-digit MAC address>
```

When using a **modification link-file**, the allowed modification codes are **'I' (Insert), 'R' (replace)** and **'D' (delete)** should be used.

When deleting a record, only the product-ID or MAC address field is mandatory.

Example:

```
I;3;H;00124B000141B633
I;2;H;00124B000141B632
I;1;H;00124B000141B634
I;1;K;00124B000141B631
D;20;;
D;8;;
D;;;00124B000141B57A
I;6;H;00124B000141B635
I;7;H;00124B000141B631
```

6.3. Middleware application (CSV-files only)

Note: It is allowed to disable the use of a middleware application. In this case, make sure the 'CSV-parser' field in the 'Data'-tab is empty. If you wish to do so, complete product and link database are allowed to be placed in the configured 'Input' folder and they must be named '.csv' and '*link*.csv' respectively. No price formatting or discount percentage calculation will be performed.*

The ESL server application can use a middleware application to process the CSV-files from the Back office into a combined CSV-file that the ESL server application uses.

The tasks of the CSV-parser are:

- Parse product database files (modification or complete databases) into a combined CSV-database containing all product info.
- Parse product-ESL link files (modification or complete databases) into a combined CSV-database containing all product-ESL links.
- (Optional) Generate a reduced article database (CSV-format) for the barcode terminal.
- (Optional) Verify price formatting and calculate the discount percentage

To process a product database, the ESL server manager will call the CSV-parser with the following parameters as specified in the setup window of the ESL server application:

```
<CSV-parser>.exe –O<Output folder> -I<Image folder>
            -i<Id field> -b<Barcode field> -d<Description field>
            -f<From price field> -s<Sell price field>
            -p<discount percentage field> -F<Field separator>
            -D<Decimal symbol>
            -Z<Display leading zero setting> (-Z –Z- or -Z0>
            -T<Create barcode terminal file (scan.csv)>
            <Database file>.csv
```

So the default call by the ESL server application would be:

```
EGF3709x.exe -O"Output" -I"Output\Images" -i1 -b2 -d3 -f5 -s6 -p7 -F; -D. -Z0
"Input\ExampleData.csv"
```

To process a product-ESL link database, the ESL server manager will call the CSV-parser with the following parameters as specified in the setup window of the ESL server application:

```
<CSV-parser>.exe –O<Output folder> -i<Id field> <*link*>.csv
```

So the default call by the ESL server application would be:

```
EGF3709x.exe -O"Output" -i1 "Input\Linked.csv"
```

If used, the record format of the truncated database file (scan.csv) for the bar code terminal that is generated will have the following format:

```
<Barcode>;<Unique ID>;<Description (optional)>
```

If no barcodes are used in your application or when the Unique ID is the barcode, then the 'Barcode'-field will be the same as the 'Unique ID'-field.

7. Barcode terminal application

A barcode terminal can be used to link an ESL (using its unique MAC addresses) to products, objects or locations. How to create these links will be described in the following chapters.

The barcode terminals that are easiest to be used with this kit is the OPH1004, OPH1005, H13, CLK3000 or the OPL9728 with a communication cradle or USB cable.

Alternatively, also the OPN2002/4/5/6 can be used for quick linking with barcodes, but place keep in mind that this terminal does not have a display for detailed feedback or a keyboard for manual editing.

It is also possible to use the H21, H22 or any other Wifi terminal to link ESL to products using Wifi. This is done by installing a local web-server on the ESL server that runs a webpage to perform the linking. See chapter 8 for more information.

7.1.　　Setting up the barcode terminal

When using the USB cable it's necessary to install Opticon's USB driver pack, which can be selected in the installer. The drivers can also be found in the 'Tools' folder of this installation.

If the 'ESL Linking Application' is not yet loaded on the terminal or needs to be updated, then the software can be loaded by using the 'Update Firmware' option in the 'Setup' window of the ESL server manager. *
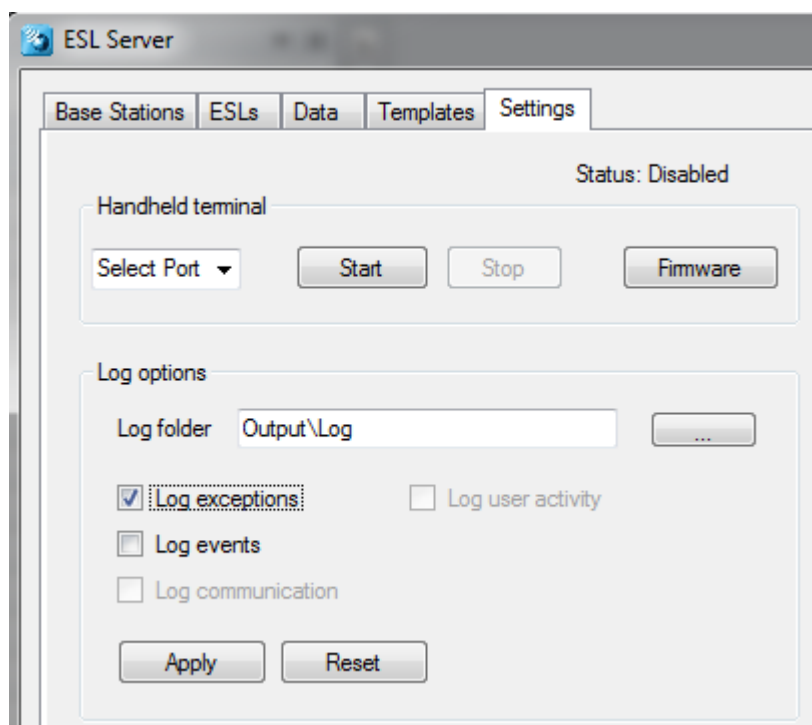
To load the software, open the 'Setup'-tab of the ESL server application. Make sure you've selected the correct COM port. Then press the 'Update Firmware' button and select the correct firmware file, being:

- 'OPH1004_H13_XPA3693x.HEX' for the OPH1004/H13
- 'OPL9728_LBF3693x.S32' for the OPL9728
- 'OPN2002_3_RFG3693x.hex' for the OPN2002 / OPN2003
- 'CLK3000_XFM36935.hex' for CLK3000
- 'OPH1005_XFP36935.hex' for OPH1005

These files can be found in the 'Firmware' folder of this installation.

* Note: Updating the firmware of the barcode terminal requires Appload to be installed. Appload can be installed by selecting 'Appload' in the installer or it can be found in the 'Tools' folder.

7.2.   Loading the database files on the terminal

Before any links can be created, the reduced product database (scan.csv) first has to be loaded on the terminal to allow the reading of the barcodes on product.

To be able to transfer files between the ESL server application and the barcode terminal the serial port of the PC is used. Make sure you've enabled and configured the correct serial port in the 'Setup'-tab in the server application.

Also verify that the field numbers of the unique ID, barcode and description match your CSV-database file (See chapter 4.1). Note that the first field is field number 0; the second is 1, etc.

Loading any files onto the terminal and sending back the resulting link database is done by placing the terminal into its cradle. As soon as the combined article database (scan.csv) or the link database has been changed, the server application will automatically synchronize the barcode terminal with the PC and process any modifications using the OseComm protocol.

7.3.   Creating links between ESLs and products

To create a link between an ESL and a product, the barcode on the back of an ESL needs to be scanned.

If the barcode on the e-paper ESLs is not accessible, it's also possible to display this same barcode on the e-paper ESLs. To do so, select all ESLs in the 'ESLs'-tab that have not yet been linked, then select the 'Show default image' action and press the 'Start' button. All ESLs should now start displaying the (default) barcode image containing the MAC address.

Note: The barcode on the ESL is truncated to fit the display, meaning that the first 6 digits (which are always the same) have been replaced by the letter 'A' or 'B'.

For re-linking and unlinking of ESLs that already show product, it's also possible add the MAC address barcode to your template file, by adding a small bar at the top of the template containing this barcode. See chapter 5.2.3 for more information.

7.3.1.   OPH100x, H13, CLK3000 or OPL9728

The first step in creating a link is pressing '1' (Link ESLs) and then selecting the used ESL variant by pressing '1' (2" e-paper) or '2' (2.7" e-paper).

The next step is scanning the MAC address barcode on the ESL, after which the terminal should sound an OK-beep and display the full MAC address.

After scanning the MAC address you can either manually enter the unique ID or scan the barcode of the product that you want to link to this ESL.

After scanning/entering the product the terminal should say that the linking was successful ('Linking successful!'). If the product wasn't found in the database it will show a message that the product wasn't found ('Product not found').

If you accidently linked the wrong product to an ESL, then you can correct this by scanning the ESL's MAC address again and linking it to the correct product.

Settings:
The settings menu of the terminal can be accessed by pressing '4' ('Settings). This menu contains 2 options:

1.   Delete data   Deletes all files from the terminal, so also all links that have not yet been sent back to the PC.
2.   Barcodes   This menu can enable/disable different barcode symbologies, so almost all barcodes can be scanned.

Version:
Shows the version of the terminal application, which can be useful for support purposes.

7.3.2. OPN2002/3/4/56

Before you start to link ESLs to products, the OPN200x first needs to know which type of ESL is going to be linked. Read one of the labels below to select the hardware model of your ESL. This action only has to be done once.

‖‖‖ ‖‖‖ Æ+H+	ESL-Type: EE200 (2" e-paper)
‖‖‖ ‖‖‖ Æ+K+	ESL-Type: EE270 (2.7" e-paper)

| ‖‖‖‖‖‖‖‖‖‖‖‖‖‖‖‖‖‖‖‖
+ – F O R M A T – + | Format Disk |
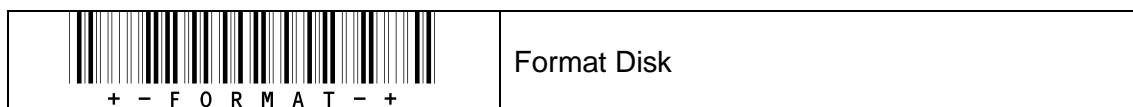
* Note: Use the format label to erase all data from the OPN200x, in case you wish to start over.

The first step to create a link is pressing the trigger button and scanning the barcode on the product.

If the product wasn't found in the database (or no database was present) it will sound an error beep to indicate the product wasn't found.

If the product was found, the laser will stay on and will start beeping every second to indicate that you can now scan MAC address of the ESL.

After scanning the MAC address an OK-beep will be given indicating that the linking was successful.

To delete an existing link, press the small button and scan either the barcode of the product or the MAC address of the ESL. The OPN200x will either sound an OK-beep or an error-beep indicating the unlinking was successful or not.

7.4. Sending the modified link-database back to the PC

After all links have been created using the barcode terminal, this information needs to be send back to the PC, so they can be processed by the server application. This can be done by placing the terminal back into its cradle (or connecting it to a USB-cable). The terminal should now automatically send the link database (linked.csv) back to the PC.

After the barcode terminal has sent back the link database, the ESL server application will call the middleware application to parse the combined database and link database into ESL image files (bitmaps) and configuration (XML) files and update the ESLs if necessary.

8. Using the Web Application for linking and monitoring

It is also possible to use the H21, H22, H32 or any other Wifi terminal to link ESL to products and monitor the ESL system using a WiFi connection. This is done by installing a local web-server on the ESL server that runs a web page to perform the linking.

During installation, make sure you've checked the option 'Install web application'. This will install the web application into the 'WebApp-Full' sub-directory of the ESL server installation.

The easiest way to set-up a local HTTP-server is by downloading and installing XAMPP from the internet and installing the following features:
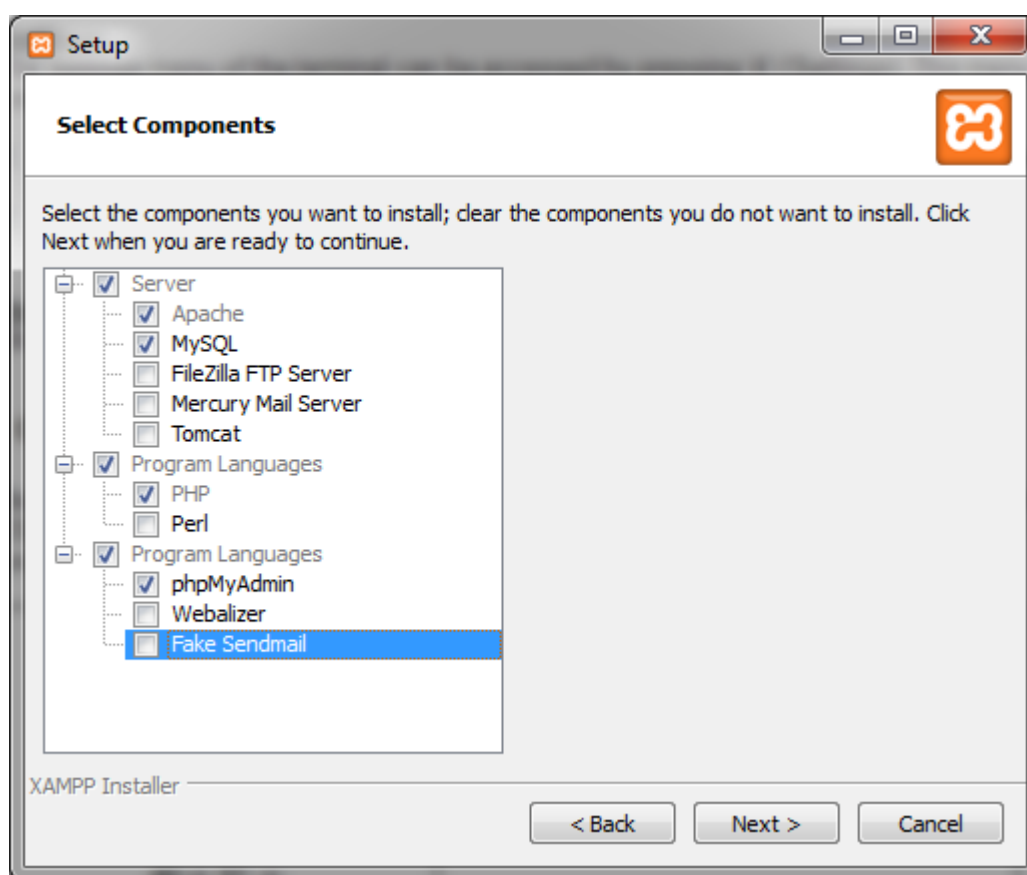
After installation, start the 'XAMPP Control panel' and press the 'Config' button next to 'Apache' and open the '**httpd.config**' file.

Change the document root directory to the '/WebApp-Full' folder of your ESL server installation by changing both occurrences of 'Documentroot':

Example:
```
#<Directory "C:/xampp/htdocs">
DocumentRoot "C:/Opticon/ESL Server/WebApp-Full"
#<Directory "C:/xampp/htdocs">
<Directory "C:/Opticon/ESL Server/WebApp-Full">
```

Save the config-file and start the 'Apache' process in the 'XAMPP Control panel'.

If you're running a different HTTP-server than Apache, please refer to the documentation of that specific HTTP-server on how to change the 'document root' directory.

The Web application should now be running. Open any web browser and enter 'localhost' to see the web application.
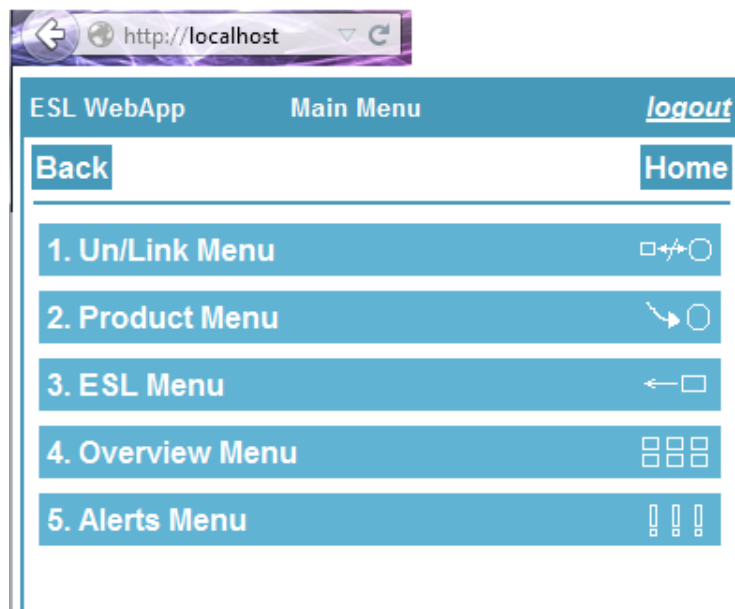
By default the Web-application is configured to use MysQL with the default MySQL settings the ESL server application.

8.1.1.   Web application with My-SQL (recommended)

To use the Web-application in combination with 'MySQL', first configure the SQL settings as described in chapter 4.2. When the ESL server is working correctly with the MySQL-database, go the 'Web application' tab, which can be found under the 'Data'-tab.

Change the document root directory of the HTTP-server to the '/WebApp-Full' folder of your ESL-server installation by changing both occurrences of 'Documentroot' as described in chapter 8.

Example:

```
DocumentRoot "C:/Opticon/ESL Server/ WebApp-Full"
<Directory "C:/Opticon/ESL Server/WebApp-Full">
```

Then make sure the 'Source folder' is pointing at the folder of the 'Web application'.

Finally hit the 'Test' button to check if the web application is working correctly.
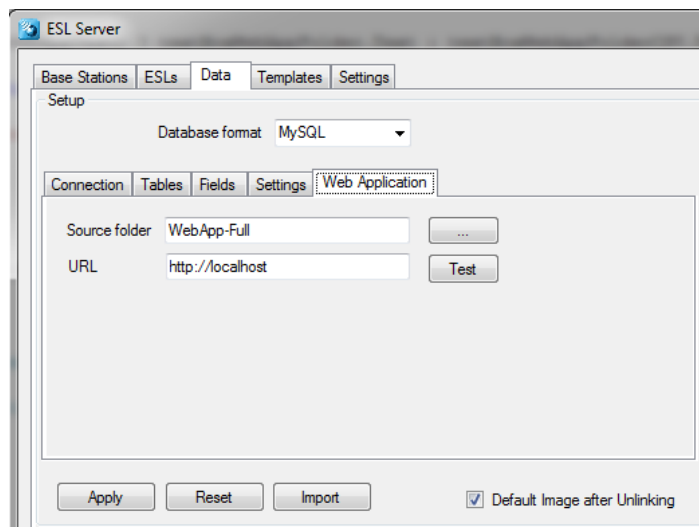
Hitting the 'Test' button automatically updates the files 'dbconfig.php' and 'config.php' in the web application folder to match your database settings.

To quickly set-up a full demo with MySQL use the query and configuration found in 'Appendix B: MySQL example query'

8.1.2. Web application with MS-SQL

To use the Web-application in combination with 'MS-SQL' is similar to MySQL, except the database format should be set to 'MS-SQL'. Currently, when using MS-SQL only linking is supported, therefore the document root should be configured to `'/WebApp-Linking'` instead of `'/WebApp-Full'`.

Be aware that running PHP in combination MS-SQL may not run work in combination with 'Apache', but should be used in combination with IIS (Internet Information Services) in order to get it to work properly with this web application. See http://php.iis.net for more information.

8.1.3. Web application with CSV (linking only)

When using CSV, then only linking and unlinking is supported by the web application.

Change the document root directory of the HTTP-server to the '`/WebApp-Linking'` folder of your ESL server installation by changing both occurrences of 'Documentroot' as described in chapter 8.

Example:

```
DocumentRoot "C:/Opticon/ESL Server/ WebApp-Linking"
<Directory "C:/Opticon/ESL Server/WebApp-Linking">
```

Then make sure the 'Source folder' is pointing to the folder of the 'Web application' as shown below.

Finally hit the 'Test' button to check if the web application is working correctly.

Hitting the 'Test' button automatically updates the files 'dbconfig.php' and 'config.php' in the web application folder to match your database settings.

8.1.4. Web application with Oracle

Currently the web application does not support the Oracle database format.

9. Running the ESL server as service

To make sure the ESL server will always run on the background and can't be closed accidently, the ESL server can also be configured to run as service.

To run the ESL server as service, go to the 'Settings-tab and enable the option: 'Run as Service'.

Secondly the Windows account details on which the ESL server has been installed need to be filled in. This makes sure that the ESL service will run under the same Windows account and will use the same configuration file when it's started manually with GUI (Graphical User Interface)

The shutdown time out of the service can be increased when closing generally takes longer than 15 seconds, i.e. when using a slow PC or when SQL is used in combination with a remote SQL server.

The service will be installed after pressing the 'Install' button and will start running as soon as the ESL server application is closed or after the PC has been rebooted.

After that, the ESL service will automatically be stopped (and restarted afterwards) when the ESL server is started with GUI.

Appendix A: Demo barcodes

The quickest way to setup an ESL demonstration with barcode terminal is to simply use the included sample database and example templates. After installation of the software and hardware, the only configuration steps necessary are finding and adding of the base station and configuring the serial port of the barcode terminal.

The example database file will automatically be send to the barcode terminal and then you can link your ESLs using the barcode terminal and the product barcodes in the images below. (See chapter 7).

Appendix B: MySQL example query and configuration

Use PhpMyAdmin to create a database called 'esl' and create 2 tables using the following queury:

```sql
-- Table structure for table `products` --

CREATE TABLE IF NOT EXISTS `products` (
  `NotUsed` varchar(1),
  `ProductId` varchar(5) NOT NULL,
  `Barcode` varchar(40) NOT NULL PRIMARY KEY,
  `Description` varchar(60),
  `Group` varchar(10),
  `StandardPrice` varchar(10),
  `SellPrice` varchar(10),
  `Discount` varchar(10),
  `Content` varchar(20),
  `Unit` varchar(10)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;

-- Table structure for table `products_staging`

CREATE TABLE IF NOT EXISTS `products_staging` (
  `NotUsed` varchar(1),
  `ProductId` varchar(5) NOT NULL,
  `Barcode` varchar(40) NOT NULL PRIMARY KEY,
  `Description` varchar(60),
  `Group` varchar(10),
  `StandardPrice` varchar(10),
  `SellPrice` varchar(10),
  `Discount` varchar(10),
  `Content` varchar(20),
  `Unit` varchar(10),
  `DELETE` varchar(1)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;

-- Example Data for table `products`

INSERT INTO `products` (`NotUsed`, `ProductId`, `Barcode`, `Description`, `Group`, `StandardPrice`,
`SellPrice`, `Discount`, `Content`, `Unit`) VALUES
('','001', '3083680012256', 'BONDUELLE CARROTS ', '93', '0,95', '5', '', '180', 'GR'),
('','002', '3083680915816', 'BONDUELLE FANTASIA FESTINI', '93', '0,99', '1,19', '', '400', 'GR'),
('','003', '3083680823609', 'BOND BUNTER MIX', '93', '1,29', '0,89', '31', '400', 'GR'),
('','004', '3083680047760', 'BOND APPEL COMPOTE', '93', '0,79', '0,79', '', '150', 'GR'),
('','005', '24000190257', 'DEL MONTE TOMATO PASTA', '93', '1,82', '1,49', '18', '400', 'GR'),
('','006', '38900014605', 'DOLE PINEAPPLE ON JUICE', '93', '1,75', '1,75', '', '680', 'GR'),
('','007', '3083680844048', 'BOND ITALIAN MIX', '93', '2,10', '2,10', '', '800', 'GR'),
('','008', '8718114412160', 'FRUCTIS 2 IN 1 SHAMPOO', '2', '3,80', '3,80', '', '300', 'ML'),
('','009', '5410091686352', 'SWARZKOPF SHAMPOO PEACH', '2', '2,95', '2,95', '', '400', 'ML'),
('','010', '4005808128600', 'NIVEA FOR MEN SPORT', '2', '3,95', '2,95', '25', '250', 'ML'),
('','011', '5410091666279', 'SWARZKOPF JUNIOR STYLING', '2', '4,50', '3,99', '11', '250', 'ML'),
('','012', '3014260000318', 'GILLETTE SHAVING FOAM', '2', '6,99', '6,99', '', '200', 'ML'),
('','013', '8710402362126', 'PRODENT TOOTHPASTE', '2', '2,05', '2,05', '', '75', 'ML'),
('','014', '57010007022464', 'AQUAFRESH TOOTHPASTE', '2', '1,80', '1,80', '', '75', 'ML'),
('','015', '8714789019468', 'COLGATE SENSATION WHITE', '2', '2,29', '2,29', '', '75', 'ML'),
('','016', '8716800452049', 'ELMEX TOOTHPASTE', '2', '1,65', '1,19', '28', '75', 'ML'),
('','017', '8711000028605', 'PICKWICK FOREST FRUIT', '27', '1,29', '1,29', '', '40', 'GR'),
('','018', '8711000018705', 'PICKWICK TEA MINT ', '27', '1,29', '1,29', '', '40', 'GR'),
('','019', '8711000008737', 'PICKWICK TEA LEMON', '27', '0,89', '0,89', '', '30', 'GR'),
('','020', '8711000003145', 'FOLGERS CLASSIC ROAST', '27', '2,99', '2,48', '17', '250', 'GR'),
('','021', '8711000194553', 'D-E SENSEO REGULAR', '27', '2,49', '2,49', '', '36', 'PCS'),
('','022', '8711000194584', 'D-E SENSEO DARK ROAST', '27', '2,49', '2,49', '', '36', 'PCS'),
('','023', '5410013142003', 'SPA & FRUIT ORANGE', '30', '0,79', '0,79', '', '1,5', 'LT'),
('','024', '5410013142004', 'SPA & FRUIT ORANGE', '30', '0,79', '0,79', '', '1,5', 'LT'),
('','025', '5410013142005', 'SPA & FRUIT ORANGE', '30', '0,79', '0,79', '', '15', 'LT'),
('','026', '8715600227284', '7-UP', '30', '0,95', '0,95', '', '1,5', 'LT'),
('','027', '5410013109679', 'SPA SPARKLING WATER', '30', '0,59', '0,59', '', '1,5', 'LT'),
('','028', '8713500008460', 'HERO CASSIS', '30', '0,79', '0,79', '', '1,25', 'LT'),
('','029', '8713300069340', 'TROPICANA APPEL JUICE', '30', '1,29', '0,99', '23', '1', 'LT'),
('','030', '2076210399997', 'HOUSE WINE WHITE DRY', '30', '3,99', '3,99', '', '2', 'LT');
```

After creating the 2 product tables, use the following configuration in the ESL server

See chapter 8 for more information on using the Web application for linking using MySQL/MS-SQL
See 'Appendix A: Demo barcodes' to demonstrate linking products using EAN/UPC barcodes

Appendix C: MS-SQL example query and configuration

Use SQL Server management studio to create a database called 'esl' and create 2 product tables using the following queury:

```sql
CREATE TABLE [products] (
  [NotUsed] varchar(1),
  [ProductId] varchar(5) NOT NULL,
  [Barcode] varchar(40) NOT NULL PRIMARY KEY,
  [Description] varchar(60),
  [Group] varchar(10),
  [StandardPrice] varchar(10),
  [SellPrice] varchar(10),
  [Discount] varchar(10),
  [Content] varchar(20),
  [Unit] varchar(10) );

CREATE TABLE [products_staging] (
  [NotUsed] varchar(1),
  [ProductId] varchar(5) NOT NULL,
  [Barcode] varchar(40) NOT NULL PRIMARY KEY,
  [Description] varchar(60),
  [Group] varchar(10),
  [StandardPrice] varchar(10),
  [SellPrice] varchar(10),
  [Discount] varchar(10),
  [Content] varchar(20),
  [Unit] varchar(10),
  [DELETE] varchar(1) );
-- Example Data for table `products`

INSERT INTO [products] ([NotUsed], [ProductId], [Barcode], [Description], [Group], [StandardPrice],
[SellPrice], [Discount], [Content], [Unit])
VALUES ('','001', '3083680012256', 'BONDUELLE CARROTS ', '93', '0,95', '0,95', '', '180', 'GR');
INSERT INTO [products] ([NotUsed], [ProductId], [Barcode], [Description], [Group], [StandardPrice],
[SellPrice], [Discount], [Content], [Unit])
VALUES ('','002', '3083680915816', 'BONDUELLE FANTASIA FESTINI', '93', '0,99', '0,99', '', '400', 'GR');
INSERT INTO [products] ([NotUsed], [ProductId], [Barcode], [Description], [Group], [StandardPrice],
[SellPrice], [Discount], [Content], [Unit])
VALUES ('','003', '3083680823609', 'BOND BUNTER MIX', '93', '1,29', '0,89', '31%', '400', 'GR');
INSERT INTO [products] ([NotUsed], [ProductId], [Barcode], [Description], [Group], [StandardPrice],
[SellPrice], [Discount], [Content], [Unit])
VALUES ('','004', '3083680047760', 'BOND APPEL COMPOTE', '93', '0,79', '0,79', '', '150', 'GR');
INSERT INTO [products] ([NotUsed], [ProductId], [Barcode], [Description], [Group], [StandardPrice],
[SellPrice], [Discount], [Content], [Unit])
VALUES ('','005', '24000190257', 'DEL MONTE TOMATO PASTA', '93', '1,82', '1,49', '18%', '400', 'GR');
INSERT INTO [products] ([NotUsed], [ProductId], [Barcode], [Description], [Group], [StandardPrice],
[SellPrice], [Discount], [Content], [Unit])
VALUES ('','006', '38900014605', 'DOLE PINEAPPLE ON JUICE', '93', '1,75', '1,75', '', '680', 'GR');
INSERT INTO [products] ([NotUsed], [ProductId], [Barcode], [Description], [Group], [StandardPrice],
[SellPrice], [Discount], [Content], [Unit])
VALUES ('','007', '3083680844048', 'BOND ITALIAN MIX', '93', '2,10', '2,10', '', '800', 'GR');
INSERT INTO [products] ([NotUsed], [ProductId], [Barcode], [Description], [Group], [StandardPrice],
[SellPrice], [Discount], [Content], [Unit])
VALUES ('','008', '8718114412160', 'FRUCTIS 2 IN 1 SHAMPOO', '2', '3,80', '3,80', '', '300', 'ML');
INSERT INTO [products] ([NotUsed], [ProductId], [Barcode], [Description], [Group], [StandardPrice],
[SellPrice], [Discount], [Content], [Unit])
VALUES ('','009', '5410091686352', 'SWARZKOPF SHAMPOO PEACH', '2', '2,95', '2,95', '', '400', 'ML');
INSERT INTO [products] ([NotUsed], [ProductId], [Barcode], [Description], [Group], [StandardPrice],
[SellPrice], [Discount], [Content], [Unit])
VALUES ('','010', '4005808128600', 'NIVEA FOR MEN SPORT', '2', '3,95', '2,95', '25%', '250', 'ML');
INSERT INTO [products] ([NotUsed], [ProductId], [Barcode], [Description], [Group], [StandardPrice],
[SellPrice], [Discount], [Content], [Unit])
```

```
VALUES ('','011', '5410091666279', 'SWARZKOPF JUNIOR STYLING', '2', '4,50', '3,99', '11%', '250', 'ML');
INSERT INTO [products] ([NotUsed], [ProductId], [Barcode], [Description], [Group], [StandardPrice],
[SellPrice], [Discount], [Content], [Unit])
VALUES ('','012', '3014260000318', 'GILLETTE SHAVING FOAM', '2', '6,99', '6,99', '', '200', 'ML');
INSERT INTO [products] ([NotUsed], [ProductId], [Barcode], [Description], [Group], [StandardPrice],
[SellPrice], [Discount], [Content], [Unit])
VALUES ('','013', '8710402362126', 'PRODENT TOOTHPASTE', '2', '2,05', '2,05', '', '75', 'ML');
INSERT INTO [products] ([NotUsed], [ProductId], [Barcode], [Description], [Group], [StandardPrice],
[SellPrice], [Discount], [Content], [Unit])
VALUES ('','014', '57010070722464', 'AQUAFRESH TOOTHPASTE', '2', '1,80', '1,80', '', '75', 'ML');
INSERT INTO [products] ([NotUsed], [ProductId], [Barcode], [Description], [Group], [StandardPrice],
[SellPrice], [Discount], [Content], [Unit])
VALUES ('','015', '8714789019468', 'COLGATE SENSATION WHITE', '2', '2,29', '2,29', '', '75', 'ML');
INSERT INTO [products] ([NotUsed], [ProductId], [Barcode], [Description], [Group], [StandardPrice],
[SellPrice], [Discount], [Content], [Unit])
VALUES ('','016', '8716800452049', 'ELMEX TOOTHPASTE', '2', '1,65', '1,65', '28%', '75', 'ML');
INSERT INTO [products] ([NotUsed], [ProductId], [Barcode], [Description], [Group], [StandardPrice],
[SellPrice], [Discount], [Content], [Unit])
VALUES ('','017', '8711000028605', 'PICKWICK FOREST FRUIT', '27', '1,29', '1,29', '', '40', 'GR');
INSERT INTO [products] ([NotUsed], [ProductId], [Barcode], [Description], [Group], [StandardPrice],
[SellPrice], [Discount], [Content], [Unit])
VALUES ('','018', '8711000018705', 'PICKWICK TEA MINT ', '27', '1,29', '1,29', '', '40', 'GR');
INSERT INTO [products] ([NotUsed], [ProductId], [Barcode], [Description], [Group], [StandardPrice],
[SellPrice], [Discount], [Content], [Unit])
VALUES ('','019', '8711000008737', 'PICKWICK TEA LEMON', '27', '0,89', '0,89', '', '30', 'GR');
INSERT INTO [products] ([NotUsed], [ProductId], [Barcode], [Description], [Group], [StandardPrice],
[SellPrice], [Discount], [Content], [Unit])
VALUES ('','020', '8711000003145', 'FOLGERS CLASSIC ROAST', '27', '2,99', '2,99', '17', '250', 'GR');
INSERT INTO [products] ([NotUsed], [ProductId], [Barcode], [Description], [Group], [StandardPrice],
[SellPrice], [Discount], [Content], [Unit])
VALUES ('','021', '8711000194553', 'D-E SENSEO REGULAR', '27', '2,49', '2,49', '', '36', 'PCS');
INSERT INTO [products] ([NotUsed], [ProductId], [Barcode], [Description], [Group], [StandardPrice],
[SellPrice], [Discount], [Content], [Unit])
VALUES ('','022', '8711000194584', 'D-E SENSEO DARK ROAST', '27', '2,49', '2,49', '', '36', 'PCS');
INSERT INTO [products] ([NotUsed], [ProductId], [Barcode], [Description], [Group], [StandardPrice],
[SellPrice], [Discount], [Content], [Unit])
VALUES ('','023', '5410013142003', 'SPA & FRUIT ORANGE', '30', '0,79', '0,79', '', '1,5', 'LT');
INSERT INTO [products] ([NotUsed], [ProductId], [Barcode], [Description], [Group], [StandardPrice],
[SellPrice], [Discount], [Content], [Unit])
VALUES ('','024', '5410013142004', 'SPA & FRUIT ORANGE', '30', '0,79', '0,79', '', '1,5', 'LT');
INSERT INTO [products] ([NotUsed], [ProductId], [Barcode], [Description], [Group], [StandardPrice],
[SellPrice], [Discount], [Content], [Unit])
VALUES ('','025', '5410013142005', 'SPA & FRUIT ORANGE', '30', '0,79', '0,79', '', '15', 'LT');
INSERT INTO [products] ([NotUsed], [ProductId], [Barcode], [Description], [Group], [StandardPrice],
[SellPrice], [Discount], [Content], [Unit])
VALUES ('','026', '8715600227284', '7-UP', '30', '0,95', '0,95', '', '1,5', 'LT');
INSERT INTO [products] ([NotUsed], [ProductId], [Barcode], [Description], [Group], [StandardPrice],
[SellPrice], [Discount], [Content], [Unit])
VALUES ('','027', '5410013109679', 'SPA SPARKLING WATER', '30', '0,59', '0,59', '', '1,5', 'LT');
INSERT INTO [products] ([NotUsed], [ProductId], [Barcode], [Description], [Group], [StandardPrice],
[SellPrice], [Discount], [Content], [Unit])
VALUES ('','028', '8713500008460', 'HERO CASSIS', '30', '0,79', '0,79', '', '1,25', 'LT');
INSERT INTO [products] ([NotUsed], [ProductId], [Barcode], [Description], [Group], [StandardPrice],
[SellPrice], [Discount], [Content], [Unit])
VALUES ('','029', '8713300069340', 'TROPICANA APPEL JUICE', '30', '1,29', '0,99', '23%', '1', 'LT');
INSERT INTO [products] ([NotUsed], [ProductId], [Barcode], [Description], [Group], [StandardPrice],
[SellPrice], [Discount], [Content], [Unit])
VALUES ('','030', '2076210399997', 'HOUSE WINE WHITE DRY', '30', '3,99', '3,99', '', '2', 'LT');
INSERT INTO [products] ([NotUsed], [ProductId], [Barcode], [Description], [Group], [StandardPrice],
[SellPrice], [Discount], [Content], [Unit])
VALUES ('','031', '2076210399993', 'HOUSE WINE RED', '30', '3,99', '3,99', '', '2', 'LT');
```

After creating the product tables, use the following configuration in the ESL server

OPTICON

Setup

Database format: Microsoft SQL

Connection | Tables | Fields | Settings | Web Application

Server address: ESL-PC\SQLEXPRESS
User ID: esl
Password: •••••••
Database: ESL

Test

Connection | Tables | Fields | Settings | Web Application

Product tables	products	Product Staging Table	products_staging
Link tables	links	Link Staging Table	links_staging
ESL status table	labelstatus	BaseStation Status Table	basestationstatus
User Table	users	Action Staging Table	actions_staging
Change Log	changelog	Action Pending Table	actions_pending
Log Messages	esllog	Action Results Table	actions_results

Connection | Tables | Fields | Settings | Web Application

Barcode	Barcode	From Price	StandardPrice
Unique ID	ProductId	Sell Price	SellPrice
Description	Description	Group	Group

Connection | Tables | Fields | Settings | Web Application

Currency: €
Culture info: en-GB

Connection | Tables | Fields | Settings | Web Application

Source folder: WebApp-Full   ...
URL: http://localhost   Test

Apply   Reset   Import   ☑ Default Image after Unlinking

See chapter 8 for more information on setting up the Web application for linking using MySQL/MS-SQL

See 'Appendix A: Demo barcodes' to demonstrate the linking of products using EAN/UPC barcodes

Appendix D: Oracle example query and configuration

Create a database called ESL and Open an SQL Worksheet in Oracle SQL developer to create 2 product tables using the following queury:

```sql
CREATE TABLE "products" (
  "NotUsed" varchar2(1),
  "ProductId" varchar2(5) NOT NULL,
  "Barcode" varchar2(40) NOT NULL PRIMARY KEY,
  "Description" varchar2(60),
  "Group" varchar2(10),
  "StandardPrice" varchar2(10),
  "SellPrice" varchar2(10),
  "Discount" varchar2(10),
  "Content" varchar2(20),
  "Unit" varchar2(10) );

CREATE TABLE "products_staging" (
  "NotUsed" varchar2(1),
  "ProductId" varchar2(5) NOT NULL,
  "Barcode" varchar2(40) NOT NULL PRIMARY KEY,
  "Description" varchar2(60),
  "Group" varchar2(10),
  "StandardPrice" varchar2(10),
  "SellPrice" varchar2(10),
  "Discount" varchar2(10),
  "Content" varchar2(20),
  "Unit" varchar2(10),
  "DELETE" varchar2(1) );

INSERT INTO "products" ("NotUsed", "ProductId", "Barcode", "Description", "Group", "StandardPrice", "SellPrice", "Discount", "Content", "Unit")
VALUES ('','001', '3083680012256', 'BONDUELLE CARROTS ', '93', '0,95', '0,95', '', '180', 'GR');
INSERT INTO "products" ("NotUsed", "ProductId", "Barcode", "Description", "Group", "StandardPrice", "SellPrice", "Discount", "Content", "Unit")
VALUES ('','002', '3083680915816', 'BONDUELLE FANTASIA FESTINI', '93', '0,99', '0,99', '', '400', 'GR');
INSERT INTO "products" ("NotUsed", "ProductId", "Barcode", "Description", "Group", "StandardPrice", "SellPrice", "Discount", "Content", "Unit")
VALUES ('','003', '3083680823609', 'BOND BUNTER MIX', '93', '1,29', '0,89', '31%', '400', 'GR');
INSERT INTO "products" ("NotUsed", "ProductId", "Barcode", "Description", "Group", "StandardPrice", "SellPrice", "Discount", "Content", "Unit")
VALUES ('','004', '3083680047760', 'BOND APPEL COMPOTE', '93', '0,79', '0,79', '', '150', 'GR');
INSERT INTO "products" ("NotUsed", "ProductId", "Barcode", "Description", "Group", "StandardPrice", "SellPrice", "Discount", "Content", "Unit")
VALUES ('','005', '24000190257', 'DEL MONTE TOMATO PASTA', '93', '1,82', '1,49', '18%', '400', 'GR');
INSERT INTO "products" ("NotUsed", "ProductId", "Barcode", "Description", "Group", "StandardPrice", "SellPrice", "Discount", "Content", "Unit")
VALUES ('','006', '38900014605', 'DOLE PINEAPPLE ON JUICE', '93', '1,75', '1,75', '', '680', 'GR');
INSERT INTO "products" ("NotUsed", "ProductId", "Barcode", "Description", "Group", "StandardPrice", "SellPrice", "Discount", "Content", "Unit")
VALUES ('','007', '3083680844048', 'BOND ITALIAN MIX', '93', '2,10', '2,10', '', '800', 'GR');
INSERT INTO "products" ("NotUsed", "ProductId", "Barcode", "Description", "Group", "StandardPrice", "SellPrice", "Discount", "Content", "Unit")
VALUES ('','008', '8718114412160', 'FRUCTIS 2 IN 1 SHAMPOO', '2', '3,80', '3,80', '', '300', 'ML');
INSERT INTO "products" ("NotUsed", "ProductId", "Barcode", "Description", "Group", "StandardPrice", "SellPrice", "Discount", "Content", "Unit")
VALUES ('','009', '5410091686352', 'SWARZKOPF SHAMPOO PEACH', '2', '2,95', '2,95', '', '400', 'ML');
INSERT INTO "products" ("NotUsed", "ProductId", "Barcode", "Description", "Group", "StandardPrice", "SellPrice", "Discount", "Content", "Unit")
VALUES ('','010', '4005808128600', 'NIVEA FOR MEN SPORT', '2', '3,95', '2,95', '25%', '250', 'ML');
INSERT INTO "products" ("NotUsed", "ProductId", "Barcode", "Description", "Group", "StandardPrice", "SellPrice", "Discount", "Content", "Unit")
VALUES ('','011', '5410091666279', 'SWARZKOPF JUNIOR STYLING', '2', '4,50', '3,99', '11%', '250', 'ML');
INSERT INTO "products" ("NotUsed", "ProductId", "Barcode", "Description", "Group", "StandardPrice", "SellPrice", "Discount", "Content", "Unit")
VALUES ('','012', '3014260000318', 'GILLETTE SHAVING FOAM', '2', '6,99', '6,99', '', '200', 'ML');
INSERT INTO "products" ("NotUsed", "ProductId", "Barcode", "Description", "Group", "StandardPrice", "SellPrice", "Discount", "Content", "Unit")
VALUES ('','013', '8710402362126', 'PRODENT TOOTHPASTE', '2', '2,05', '2,05', '', '75', 'ML');
```

```
INSERT INTO "products" ("NotUsed", "ProductId", "Barcode", "Description", "Group", "StandardPrice",
"SellPrice", "Discount", "Content", "Unit")
VALUES ('','014', '5701007022464', 'AQUAFRESH TOOTHPASTE', '2', '1,80', '1,80', '', '75', 'ML');
INSERT INTO "products" ("NotUsed", "ProductId", "Barcode", "Description", "Group", "StandardPrice",
"SellPrice", "Discount", "Content", "Unit")
VALUES ('','015', '8714789019468', 'COLGATE SENSATION WHITE', '2', '2,29', '2,29', '', '75', 'ML');
INSERT INTO "products" ("NotUsed", "ProductId", "Barcode", "Description", "Group", "StandardPrice",
"SellPrice", "Discount", "Content", "Unit")
VALUES ('','016', '8716800452049', 'ELMEX TOOTHPASTE', '2', '1,65', '1,65', '28%', '75', 'ML');
INSERT INTO "products" ("NotUsed", "ProductId", "Barcode", "Description", "Group", "StandardPrice",
"SellPrice", "Discount", "Content", "Unit")
VALUES ('','017', '8711000028605', 'PICKWICK FOREST FRUIT', '27', '1,29', '1,29', '', '40', 'GR');
INSERT INTO "products" ("NotUsed", "ProductId", "Barcode", "Description", "Group", "StandardPrice",
"SellPrice", "Discount", "Content", "Unit")
VALUES ('','018', '8711000018705', 'PICKWICK TEA MINT ', '27', '1,29', '1,29', '', '40', 'GR');
INSERT INTO "products" ("NotUsed", "ProductId", "Barcode", "Description", "Group", "StandardPrice",
"SellPrice", "Discount", "Content", "Unit")
VALUES ('','019', '8711000008737', 'PICKWICK TEA LEMON', '27', '0,89', '0,89', '', '30', 'GR');
INSERT INTO "products" ("NotUsed", "ProductId", "Barcode", "Description", "Group", "StandardPrice",
"SellPrice", "Discount", "Content", "Unit")
VALUES ('','020', '8711000003145', 'FOLGERS CLASSIC ROAST', '27', '2,99', '2,99', '17', '250', 'GR');
INSERT INTO "products" ("NotUsed", "ProductId", "Barcode", "Description", "Group", "StandardPrice",
"SellPrice", "Discount", "Content", "Unit")
VALUES ('','021', '8711000194553', 'D-E SENSEO REGULAR', '27', '2,49', '2,49', '', '36', 'PCS');
INSERT INTO "products" ("NotUsed", "ProductId", "Barcode", "Description", "Group", "StandardPrice",
"SellPrice", "Discount", "Content", "Unit")
VALUES ('','022', '8711000194584', 'D-E SENSEO DARK ROAST', '27', '2,49', '2,49', '', '36', 'PCS');
INSERT INTO "products" ("NotUsed", "ProductId", "Barcode", "Description", "Group", "StandardPrice",
"SellPrice", "Discount", "Content", "Unit")
VALUES ('','023', '5410013142003', 'SPA FRUIT ORANGE', '30', '0,79', '0,79', '', '1,5', 'LT');
INSERT INTO "products" ("NotUsed", "ProductId", "Barcode", "Description", "Group", "StandardPrice",
"SellPrice", "Discount", "Content", "Unit")
VALUES ('','024', '5410013142004', 'SPA FRUIT ORANGE', '30', '0,79', '0,79', '', '1,5', 'LT');
INSERT INTO "products" ("NotUsed", "ProductId", "Barcode", "Description", "Group", "StandardPrice",
"SellPrice", "Discount", "Content", "Unit")
VALUES ('','025', '5410013142005', 'SPA FRUIT ORANGE', '30', '0,79', '0,79', '', '15', 'LT');
INSERT INTO "products" ("NotUsed", "ProductId", "Barcode", "Description", "Group", "StandardPrice",
"SellPrice", "Discount", "Content", "Unit")
VALUES ('','026', '8715600227284', '7-UP', '30', '0,95', '0,95', '', '1,5', 'LT');
INSERT INTO "products" ("NotUsed", "ProductId", "Barcode", "Description", "Group", "StandardPrice",
"SellPrice", "Discount", "Content", "Unit")
VALUES ('','027', '5410013109679', 'SPA SPARKLING WATER', '30', '0,59', '0,59', '', '1,5', 'LT');
INSERT INTO "products" ("NotUsed", "ProductId", "Barcode", "Description", "Group", "StandardPrice",
"SellPrice", "Discount", "Content", "Unit")
VALUES ('','028', '8713500008460', 'HERO CASSIS', '30', '0,79', '0,79', '', '1,25', 'LT');
INSERT INTO "products" ("NotUsed", "ProductId", "Barcode", "Description", "Group", "StandardPrice",
"SellPrice", "Discount", "Content", "Unit")
VALUES ('','029', '8713300069340', 'TROPICANA APPEL JUICE', '30', '1,29', '0,99', '23%', '1', 'LT');
INSERT INTO "products" ("NotUsed", "ProductId", "Barcode", "Description", "Group", "StandardPrice",
"SellPrice", "Discount", "Content", "Unit")
VALUES ('','030', '2076210399997', 'HOUSE WINE WHITE DRY', '30', '3,99', '3,99', '', '2', 'LT');
INSERT INTO "products" ("NotUsed", "ProductId", "Barcode", "Description", "Group", "StandardPrice",
"SellPrice", "Discount", "Content", "Unit")
 VALUES ('','031', '2076210399993', 'HOUSE WINE RED', '30', '3,99', '3,99', '', '2', 'LT');
```

Appendix E: Using the SQL API

(My)SQL can be used as API to create your own application by executing SQL queries to control and monitor the ESL system, while running the ESL server as background service (See chapter 9 on how run the ESL server as service).

See chapter 4.2 for information on how to use the product and link staging tables to update ESLs and status tables to monitor the ESL system remotely.

Besides the SQL tables described in chapter 4.2, the ESL server also automatically creates the following 3 SQL tables to allow control over the ESL server using SQL queries.

- **Actions_staging**: Staging table used to insert new actions that need to be executed.

ACTION	varchar(32)	utf8_general_ci	Geen	🔑
MAC	varchar(16)	utf8_general_ci	Geen	🔑
USER	varchar(64)	utf8_general_ci	NULL	
PARAM1	varchar(32)	utf8_general_ci	NULL	
PARAM2	varchar(32)	utf8_general_ci	NULL	
PARAM3	varchar(32)	utf8_general_ci	NULL	
TIME	datetime		NULL	

- **Actions_pending**: Actions that involve commands that need to be send to ESLs are moved from the staging table to this table to indicate that the command is queued and will be sent as soon as the ESL wakes up.

ACTION	varchar(32)	utf8_general_ci	Geen	🔑
MAC	varchar(16)	utf8_general_ci	Geen	🔑
USER	varchar(64)	utf8_general_ci	NULL	
PARAM1	varchar(32)	utf8_general_ci	NULL	
PARAM2	varchar(32)	utf8_general_ci	NULL	
PARAM3	varchar(32)	utf8_general_ci	NULL	
TIME	datetime		NULL	

- **Actions_results**: All actions that have immediate result will be moved from the staging table to this table. This includes base station commands and failed ESL commands.

ACTION	varchar(32)	utf8_general_ci	Geen	🔑
MAC	varchar(16)	utf8_general_ci	Geen	🔑
USER	varchar(64)	utf8_general_ci	NULL	
PARAM1	varchar(32)	utf8_general_ci	NULL	
PARAM2	varchar(32)	utf8_general_ci	NULL	
PARAM3	varchar(32)	utf8_general_ci	NULL	
TIME	datetime		NULL	
RESULT	varchar(256)	utf8_general_ci	NULL	

Supported ESL actions

The following list of SQL-commands is currently supported to initiate an ESL action. Valid commands will be moved to the action_pending table. Disgarded commands will be moved to the action_results table

ACTION	MAC	PARAM1
RESET	MAC-Address (16)	
`INSERT INTO `actions_staging`(`ACTION`, `MAC`) VALUES ('RESET', '3889DC000000A842');`		
DEFAULT_IMAGE	MAC-Address (16)	
`INSERT INTO `actions_staging`(`ACTION`, `MAC`) VALUES ('DEFAULT_IMAGE', '3889DC000000A842');`		
REFRESH_IMAGE	MAC-Address (16)	
`INSERT INTO `actions_staging`(`ACTION`, `MAC`) VALUES ('REFRESH_IMAGE', '3889DC000000A842');`		
RESEND_IMAGE	MAC-Address (16)	
`INSERT INTO `actions_staging`(`ACTION`, `MAC`) VALUES ('RESEND_IMAGE', '3889DC000000A842');`		
REQUEST_SETTINGS	MAC-Address (16)	
`INSERT INTO `actions_staging`(`ACTION`, `MAC`) VALUES ('REQUEST_SETTINGS', '3889DC000000A842');`		
UNLINK	MAC-Address (16)	
`INSERT INTO `actions_staging`(`ACTION`, `MAC`) VALUES ('UNLINK', '3889DC000000A842');`		
REASSOCIATE	MAC-Address (16)	
`INSERT INTO `actions_staging`(`ACTION`, `MAC`) VALUES ('REASSOCIATE', '3889DC000000A842');`		
KICK_FROM_EBS	MAC-Address (16)	
`INSERT INTO `actions_staging`(`ACTION`, `MAC`) VALUES ('KICK_FROM_EBS', '3889DC000000A842');`		
MOVE_TO_EBS	MAC-Address (16)	MAC-Address EBS (12)
`INSERT INTO `actions_staging`(`ACTION`, `MAC`, `PARAM1`) VALUES ('MOVE_TO_EBS', '3889DC000000A842', '00126AF5001234');`		
SET_LANID	MAC-Address (16)	
`INSERT INTO `actions_staging`(`ACTION`, `MAC`) VALUES ('SET_LANID', '3889DC000000A842');`		
CLEAR_LANID	MAC-Address (16)	
`INSERT INTO `actions_staging`(`ACTION`, `MAC`) VALUES ('CLEAR_LANID', '3889DC000000A842');`		
DEACTIVATE	MAC-Address (16)	
`INSERT INTO `actions_staging`(`ACTION`, `MAC`) VALUES ('DEACTIVATE', '3889DC000000A842');`		
FACTORY_DEFAULT	MAC-Address (16)	
`INSERT INTO `actions_staging`(`ACTION`, `MAC`) VALUES ('FACTORY_DEFAULT', '3889DC000000A842');`		
CHANGE_MAC_ADDRESS	MAC-Address (16)	MAC-Address (new) (16) (For factory only)
`INSERT INTO `actions_staging`(`ACTION`, `MAC`, `PARAM1`) VALUES ('CHANGE_MAC_ADDRESS', '00124B00049F203D', '3889DC000000A000');`		

Supported ESL actions

The following list of SQL-commands is currently supported to change ESL settings. Valid commands will be moved to the action_pending table. Disgarded settings and settings that match the current setting will be moved to the action_results table.

ACTION	MAC	PARAM1 (default; min; max)	
POLL_INTERVAL (int)	MAC-Address (16)	Min. poll interval in seconds (**20**, 5, 200)	
INSERT INTO `actions_staging`(`ACTION`, `MAC`, `PARAM1`) VALUES ('POLL_INTERVAL', '3889DC000000A842', 20);			
POLL_TIMEOUT (int)	MAC-Address (16)	Poll timeout in polls (**10**, 5, 30)	
INSERT INTO `actions_staging`(`ACTION`, `MAC`, `PARAM1`) VALUES ('POLL_TIMEOUT', '3889DC000000A842', 20);			
SCAN_INTERVAL (int)	MAC-Address (16)	Associate interval in seconds (**60**, 60, 300)	
INSERT INTO `actions_staging`(`ACTION`, `MAC`, `PARAM1`) VALUES ('SCAN_INTERVAL', '3889DC000000A842', 60);			
DISPLAY_OPTIONS (int)	MAC-Address (16)	**0: normal**, 1: upside down	
INSERT INTO `actions_staging`(`ACTION`, `MAC`, `PARAM1`) VALUES ('DISPLAY_OPTIONS', '3889DC000000A842', 1);			
SCAN_CHANNELS (flags)	MAC-Address (16)	**11: 0x00000800** 12: 0x00001000 13: 0x00002000 14: 0x00004000 **15: 0x00008000** 16: 0x00010000 17: 0x00020000 18: 0x00040000	19: 0x00080000 **20: 0x00100000** 21: 0x00200000 22: 0x00400000 23: 0x00800000 24: 0x01000000 **25: 0x02000000** **26: 0x04000000**
INSERT INTO `actions_staging`(`ACTION`, `MAC`, `PARAM1`) VALUES ('SCAN_CHANNELS', '3889DC000000A842', '101746688'); (0x06108800 = 11, 15, 20, 25, 26)			
TEMPERATURE	MAC-Address (16)	Temperature range. (0=default; 0, 4) (>0 values are to compensate for lower temperatures, but shorten battery life)	
INSERT INTO `actions_staging`(`ACTION`, `MAC`, `PARAM1`) VALUES ('TEMPERATURE', '3889DC000000A842', 0);			

Supported Base station actions

The following list of SQL-commands is currently supported to initiate a base-station-action.
Successfully executed and discarded commands will be moved to the action results-table.

ACTION	MAC	PARAM1
REACTIVATE_START	MAC-Address (12)	Timeout in sec (min 30, max 900)
`INSERT INTO `actions_staging`(`ACTION`, `MAC`, `PARAM1`) VALUES ('REACTIVATE_START', '00126AF5FFF8', 180);`		
REACTIVATE_STOP	MAC-Address (12)	
`INSERT INTO `actions_staging`(`ACTION`, `MAC`) VALUES ('REACTIVATE_STOP', '00126AF5FFF8');`		
DEACTIVATE	Start range (MAC-Address (16)	End range (MAC-Address 16)
`INSERT INTO `actions_staging`(`ACTION`, `MAC`, `PARAM1`) VALUES ('DEACTIVATE', '3889DC0001000000', '3889DC000100FFFF');`		
DISCOVER		
`INSERT INTO `actions_staging`(`ACTION`, `MAC`) VALUES ('DISCOVER', '');`		
BALANCE_ESLS		*(Use preferably once a day)*
`INSERT INTO `actions_staging`(`ACTION`, `MAC`) VALUES ('BALANCE_ESLS', '');`		
RESET	MAC-Address (12)	
`INSERT INTO `actions_staging`(`ACTION`, `MAC`) VALUES ('RESET', '00126AF5FFF8');`		
ADD	MAC-Address (12)	
`INSERT INTO `actions_staging`(`ACTION`, `MAC`) VALUES ('ADD', '00126AF5FFF8');`		
REMOVE		
`INSERT INTO `actions_staging`(`ACTION`, `MAC`) VALUES ('REMOVE', '00126AF5FFF8');`		
FIRMWARE_UPDATE		
`INSERT INTO `actions_staging`(`ACTION`, `MAC`, `PARAM1`) VALUES ('FIRMWARE_UPDATE', '00126AF5FFF8', 'EBS40-IBUV0052.bin');`		

Supported Base station settings

The following list of SQL-commands are currently supported to change a base-station-setting.
Successfully executed and disgarded commands will be moved to the action results-table.

ACTION	MAC	PARAM1
CHANNEL	MAC-Address (12)	RF-Channel (int) (**default 11**, min 11, max 26)
`INSERT INTO `actions_staging`(`ACTION`, `MAC`, `PARAM1`) VALUES ('CHANNEL','00126AF5FFF8',15);`		
PANID	MAC-Address (12)	PAN-ID (4) (default: last 4 digits of MAC)
`INSERT INTO `actions_staging`(`ACTION`, `MAC`, `PARAM1`) VALUES ('PANID','00126AF5FFF8','12AB');`		
LANID	MAC-Address (12)	LAN-ID (4) (**default 0000**, min 0000, max FFFE)
`INSERT INTO `actions_staging`(`ACTION`, `MAC`, `PARAM1`) VALUES ('LANID','00126AF5FFF8','0003');`		
LOCAL_NAME	MAC-Address (12)	LAN-ID (4) (**default 0000**, min 0000, max FFFE)
`INSERT INTO `actions_staging`(`ACTION`, `MAC`, `PARAM1`) VALUES ('LOCAL_NAME','00126AF5FFF8','FIRST_FLOOR');`		

Appendix F: Version history

V0.1	First preliminary release	October 11, 2011
V0.2	Added following features: - OPL9728 compatibility (LFD36931) - 2.7" e-paper compatibility - Support of alphanumeric product IDs Fixed adding images to image templates Fixed delete database button Updated manual to match Opticon style	November 10, 2011
	Fixed cross-references	November 18, 2011
V1.0	Fully updated manual to support: - EBS-30 base station - New ESL server application - OPN2002 compatibility	January 3, 2013
V1.14	Stability and performance updates Added 4.4" and 7.4" e-paper compatibility Added import settings/templates feature Converted databases to UTF-8 for improved language support Updated base station and ESL firmware for improved stability and performance	June 18, 2013
V1.20	Added following features: - EE201 support, MySQL & MS-SQL support	August 8, 2013
V1.21	Added following features: - Fixes related to MySQL and MS-SQL support - Enabled use of SQL status table 'LabelStatus' to provide feedback on the ESL system - Improved performance with large link and product databases - Improved performance with large quantities of ESLs - Added support of rotated text, barcodes and images to Image templates	August 16, 2013
V1.22-V.129	- Fixes related to MySQL, MS-SQL and CSV support - Improved performance with large link and product databases - Improved performance with large quantities of ESLs - Various bug fixes - Updated firmware for EBS30 and EE200/1 - Fixed Windows explorer crash - Added multi-language support - Added 2D barcode support	February 11, 2014
V1.30	- Extended UAC and Monitoring - Added minimize to tray options - Improved performance with large quantities of ESLs - Updated firmware for EBS30	March 27, 2014

	- Fixed progress bar on closing application - Fixed minor language issues - Update manual	
V1.31	- Updated firmware for EBS30 - Added CSV/SQL hybrid solution - Added external content management solution - Minor bug fixes for image updating and CSV-parsing - Fixed issue of linked ESLs that weren't listed in the ESL-tab under certain conditions - Added Web Application to installer and documentation	April 7, 2014
V1.32	- Added OPH1005 link application - Added French language support - Added Italian language support - Minor authentication bug fix	April 11, 2014
V1.33- V1.34	- Added protective measures against loosing of templates, links or database files - Enabled logging of user activity by default - Prevented double execution of ESL server - Fixes various rare exceptions - Fixed issue with image id's not incrementing - Made SQL status/logging tables configurable - Fixed not automatically reloading of Product table after using SQL staging tables - Option to show default image on unlinking - Automatically change CSV-log file name	April 30, 2014
V1.35	- Improvements to CSV-parser - Added option show default image after unlinking - Fixed crash on removing USB-cable - Fixed rare image update issues - Fixed reactivation problems (requires EBS3x firmware IBMV0014a or higher)	May 6, 2014
V1.36	- Fixed issue with image IDs not incrementing on re-linking - Fixed crash related to OseComm / USB - Fixed SQL issue with removing products - Moved processing of Link files from CSV-parser to ESL-server - Resolved rare exceptions on logging - Added support of 'unlink' option in ESL-tab - Improved importing of database/link files - Fixed resending of images to unlinked ESLs	May 9, 2014

V1.37	- Fixed issue with disconnected EBS3x - Fixed SQL logging date/time issue	May 15, 2014
V1.38	- Improved 4.4" (EE440) support - Fixed logging failed pop-up (CSV only) - Fixed deleting of products (SQL only) - Improved handling of unreliable / remote SQL connections - Added preliminary support for running parallel ESL networks	May 27, 2014
V1.39-1.40	- Improved SQL performance and stability - Added 'Find best base station' action - Updated EE20x firmware with LAN-ID support and improved searching algorithm to find best base station connection. - Updated EBS3x firmware with LAN-ID support - Improved firmware upgrading of EE201 (variant 'IBIxxxx') - Added EE202 support (=EE201 with new display)	June 30, 2014
V1.41	- Added option to run and ESL server as Window service. - Fixed start-up message base station table - Fixed out-of-sync SQL tables after Ethernet connection issues	July 9, 2014
V1.42	- **Added graphical template designer** - Lowered CPU usage - Add (non-filled) rectangles and ellipses to template designer - Added shrink-to-fit option for texts	July 30, 2014
V1.43	- Updated EBS3x firmware (IBMV0016) with stability fixes - Improvements to graphical template designer - Improved EBS3x firmware updating	August 5, 2014
V1.45	- Added new filter option: 'Not contain' - Added automatic / manual load balancing options (see chapter 3.6.2) - Allow modification databases with no CSV-parser configured - Improved link quality indication. Packet loss is now included in the calculation of the link quality. (See chapter 3.4)	August 22, 2014
V1.46.2	- Improvements to barcode terminal applications (incl. SQL-support, OPH3000 compatibility) - Fixed issues with 'Show default image on unlinking' - Simplified the configuration of SQL - Added example SQL queries to the Appendixes - Improved UTF8 support on copying example data to template editor - Fixed issue with 'Link to Data' not working - Added compatibility with EBS40	October 16, 2014

V1.47.0	- Updated SQL solution with Web application with (un)linking, user access control, product reviewing, price editing and monitoring - Added change-log table to SQL - Added 'users' table to SQL for UAC - Added support of different e-paper formats (1.5", 2.9" and 4.2") - Added possibility to import CSV-file into an SQL product table	December 24, 2014
V1.47.2	- Added support of tab-separated files - Improved firmware update of EBS3x/EBS4x - Fixed missing file in new web-application	January 22, 2015
V1.48.0	- Added Oracle database support - Added shop selection interface to web application - Improved Asian language support for MySQL - Fixed EAN13 encoding bug - Fixed issue with link files not being processed - Improved firmware updating of EBSxx	March 23, 2015
V1.48.5	- Added preliminary EE580 support - Resolved issues with duplicate base station in base station list - Added check for IP-address change of base station at start-up (still recommended to use static DHCP leases) - Added field numbers to example data text boxes for easy-of-use - Automatically configure web application settings on changing database settings in ESL-server - Added SQL API for executing actions and changing settings in ESL server - Fixed incorrect value of field separator - Fixed missing primary key exception at start-up - Fixed range field in ESL-form - Improved ESL firmware updating	May 28, 2015
V1.48.14	- Changed battery low threshold for all AAA based ESLs to 2.35V - Added automatic ESL firmware updating by placing a single firmware file in input-folder - Fixed exceptions on unknown barcode/unique SQL-columns - Improved unlinking when using external content management solution - Added time stamp support in templates - Improved image status in ESL-form - Added preliminary EE580R/Y support (with Red/Yellow) - Fixed issue with the handling of the product staging tables - Bugfix: image update error after 255 updates - Added link-action to ESL form	July 28, 2015
V1.49.1	- Simplified user interface of template editor - Fixed bug in template editor causing template selecting file to become corrupt - Improvements to importing (SQL-)settings - Added logging of user-actions when using SQL-action	September 4, 2015

	tables - Extended web application with new features (SQL only) - Added new EE440 firmware (lowers power consumption) - Added new EE20x firmware (reintroduces battery-voltage) - Added new EBS3x firmware (adds version to web-page / reports battery-voltages of known ESLs at start-up) - Added new EBS40 firmware (fixes fallback to default-IP with DHCP enabled / reports battery-voltages of known ESLs at start-up) - Fixed rare "ESL Server already running"-issue - Added temperature range as measured by the EE201/2 (requires firmware IBxV0070 or higher) - Added new IMAGE_STATUS value: ERROR (image file could not be generated, read or found)	